Netezza ODBC, JDBC, and OLE DB Installation and Configuration Guide

Document Number: 20751-12 Rev. 5
Software Release: 6.0.2
Revised: May 3, 2011

Netezza Corporation

Corporate Headquarters
26 Forest St., Marlborough, Massachusetts 01752
tel 508.382.8200 fax 508.382.8300 www.netezza.com

The specifications and information regarding the products described in this manual are subject to change without notice. All statements, information, and recommendations in this manual are believed to be accurate.

Netezza makes no representations or warranties of any kind, express or implied, including, without limitation, those of merchantability, fitness for a particular purpose, and non infringement, regarding this manual or the products' use or performance. In no event will Netezza be liable for indirect, incidental, consequential, special, or economic damages (including lost business profits, business interruption, loss or damage of data, and the like) arising out of the use or inability to use this manual or the products, regardless of the form of action, whether in contract, tort (including negligence), breach of warranty, or otherwise, even if Netezza has been advised of the possibility of such damages.

Netezza, the Netezza logo, the circle-N logo, TwinFin, Skimmer, Snippet Blades, S-Blades, NPS, Snippet, Snippet Processing Unit, SPU, Snippet Processing Array, SPA, Performance Server, Netezza Performance Server, Asymmetric Massively Parallel Processing, AMPP, Intelligent Query Streaming and other marks are trademarks or registered trademarks of Netezza Corporation in the United States and/or other countries. All rights reserved.

Red Hat is a trademark or registered trademark of Red Hat, Inc. in the United States and/or other countries.

Linux is a trademark or registered trademark of Linus Torvalds in the United States and/or other countries.

D-CC, D-C++, Diab+, FastJ, pSOS+, SingleStep, Tornado, VxWorks, Wind River, and the Wind River logo are trademarks, registered trademarks, or service marks of Wind River Systems, Inc. Tornado patent pending.

APC and the APC logo are trademarks or registered trademarks of American Power Conversion Corporation.

All document files and software of the above named third-party suppliers are provided "as is" and may contain deficiencies. Netezza and its suppliers disclaim all warranties of any kind, express or implied, including, without limitation, those of merchantability, fitness for a particular purpose, and non infringement.

In no event will Netezza or its suppliers be liable for indirect, incidental, consequential, special, or economic damages (including lost business profits, business interruption, loss or damage of data, and the like), or the use or inability to use the above-named third-party products, even if Netezza or its suppliers have been advised of the possibility of such damages.

All other trademarks mentioned in this document are the property of their respective owners.

Document Number: 20751-12 Rev. 5

Software Release Number: 6.0.2

Netezza ODBC, JDBC, and OLE DB Installation and Configuration Guide

Copyright © 2001-2011 Netezza Corporation.

All rights reserved.

PostgreSQL

Portions of this publication were derived from PostgreSQL documentation. For those portions of the documentation that were derived originally from PostgreSQL documentation, and only for those portions, the following applies:

PostgreSQL is copyright © 1996-2001 by the PostgreSQL global development group and is distributed under the terms of the license of the University of California below.

Postgres95 is copyright © 1994-5 by the Regents of the University of California.

Permission to use, copy, modify, and distribute this documentation for any purpose, without fee, and without a written agreement is hereby granted, provided that the above copyright notice and this paragraph and the following two paragraphs appear in all copies.

In no event shall the University of California be liable to any party for direct, indirect, special, incidental, or consequential damages, including lost profits, arising out of the use of this documentation, even if the University of California has been advised of the possibility of such damage.

The University of California specifically disclaims any warranties, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. The documentation provided hereunder is on an "as-is" basis, and the University of California has no obligations to provide maintenance, support, updates, enhancements, or modifications.

ICU Library

The Netezza implementation of the ICU library is an adaptation of an open source library Copyright (c) 1995-2003 International Business Machines Corporation and others.

ICU License - ICU 1.8.1 and later
COPYRIGHT AND PERMISSION NOTICE

Copyright (c) 1995-2003 International Business Machines Corporation and others
All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, provided that the above copyright notice(s) and this permission notice appear in all copies of the Software and that both the above copyright notice(s) and this permission notice appear in supporting documentation.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR HOLDERS INCLUDED IN THIS NOTICE BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

Except as contained in this notice, the name of a copyright holder shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization of the copyright holder.

ODBC Driver

The Netezza implementation of the ODBC driver is an adaptation of an open source driver, Copyright © 2000, 2001, Great Bridge LLC. The source code for this driver and the object code of any Netezza software that links with it are available upon request to source-request@netezza.com

Botan License

Copyright (C) 1999-2008 Jack Lloyd

2001 Peter J Jones
2004-2007 Justin Karneges
2005 Matthew Gregan
2005-2006 Matt Johnston
2006 Luca Piccarreta
2007 Yves Jerschow
2007-2008 FlexSecure GmbH
2007-2008 Technische Universitat Darmstadt
2007-2008 Falko Strenzke
2007-2008 Martin Doering
2007 Manuel Hartl
2007 Christoph Ludwig
2007 Patrick Sona

All rights reserved.

Redistribution and use in source and binary forms, for any use, with or without modification, of Botan (http://botan.randombit.net/license.html) is permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions, and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions, and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR(S) "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE, ARE DISCLAIMED.

IN NO EVENT SHALL THE AUTHOR(S) OR CONTRIBUTOR(S) BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITYOF SUCH DAMAGE.

Regulatory Notices

Install the NPS system in a restricted-access location. Ensure that only those trained to operate or service the equipment have physical access to it. Install each AC power outlet near the NPS rack that plugs into it, and keep it freely accessible.

Provide approved 30A circuit breakers on all power sources.

Product may be powered by redundant power sources. Disconnect ALL power sources before servicing.

High leakage current. Earth connection essential before connecting supply. Courant de fuite élevé. Raccordement à la terre indispensable avant le raccordement au réseau.

Homologation Statement

Attention: This product is not intended to be connected directly or indirectly by any means whatsoever to interfaces of public telecommunications networks nor is it to be used in a Public Services Network.

FCC - Industry Canada Statement

This equipment has been tested and found to comply with the limits for a Class A digital device, pursuant to part 15 of the FCC rules. These limits are designed to provide reasonable protection against harmful interference when the equipment is operated in a commercial environment. This equipment generates, uses, and can radiate radio-frequency energy and, if not installed and used in accordance with the instruction manual, may cause harmful interference to radio communications. Operation of this equipment in a residential area is likely to cause harmful interference, in which case users will be required to correct the interference at their own expense.

This Class A digital apparatus meets all requirements of the Canadian Interference-Causing Equipment Regulations.

Cet appareil numérique de la classe A respecte toutes les exigences du Règlement sur le matériel brouilleur du Canada.

WEEE

Netezza Corporation is committed to meeting the requirements of the European Union (EU) Waste Electrical and Electronic Equipment (WEEE) Directive. This Directive requires producers of electrical and electronic equipment to finance the takeback, for reuse or recycling, of their products placed on the EU market after August 13, 2005.

CE Statement (Europe)

This product complies with the European Low Voltage Directive 73/23/EEC and EMC Directive 89/336/EEC as amended by European Directive 93/68/EEC.

Warning: This is a class A product. In a domestic environment this product may cause radio interference in which case the user may be required to take adequate measures.

VCCI Statement

この装置は、情報処理装置等電波障害自主規制協議会 (VCCI) の基準に基づくクラス A 情報技術装置です。この装置を家庭環境で使用すると電波妨害を引き起越すことがあります。この場合には使用者が適切な対策を講ずるう要求されることがあります。

Table of Contents

Preface

1 Introduction

Background 1-1
Architecture 1-2
ODBC 1-2
JDBC 1-3
Process 1-4
ODBC 1-4
JDBC 1-4
OLE DB 1-5
Installation Files Access 1-5

2 Support and Release Compatibility

Supported Platforms 2-1
Release Compatibility Matrix 2-2
ODBC Driver Version Support 2-3
ODBC Standards Conformance and API Versions 2-3
JDBC Driver Data Source Support 2-4

3 Installing and Uninstalling Client Tools

Windows Client Tools Installation 3-1
Created Subfolders 3-2
Client Tools Installation Procedure for Windows 3-2
UNIX/Linux Client Tools Installation 3-3
Created Subdirectories 3-3
Client Tools Installation Procedure for UNIX/Linux 3-4
Uninstalling Client Tools for Windows 3-4
Uninstalling Client Tools for UNIX/Linux 3-5

4 ODBC Drivers and Driver Managers

ODBC Drivers 4-1
ODBC Driver Functions 4-1
ODBC Driver Properties 4-2
ODBC Driver Managers 4-2

ODBC Driver Manager Functions 4-3
Windows Driver Managers 4-3
Linux/UNIX Driver Managers 4-3
unixODBC Driver Manager Installation 4-4
DataDirect Driver Manager Installation 4-5

5 Installing and Configuring ODBC

Installing and Configuring ODBC for Windows 5-1
ODBC Driver Installation for Windows via Command Line 5-1
ODBC Driver Installation Steps for Windows via the GUI 5-2
ODBC Driver Installation Procedure for Windows via the GUI 5-2
ODBC Driver Data Source Configuration for Windows 5-3
Configuring ODBC for UNIX/Linux 5-10
ODBC Driver Configuration for UNIX/Linux 5-10
ODBC Data Source Configuration for UNIX/Linux 5-12
ODBC Connectivity Validation for UNIX/Linux 5-16

6 Installing and Configuring JDBC

Installing JDBC for Windows 6-1
JDBC Driver Installation Steps for Windows 6-1
JDBC Driver Installation Procedure for Windows via the GUI 6-1
Installing JDBC for UNIX/Linux 6-2
JDBC Driver Installation for UNIX/Linux on a Supported Platform 6-2
JDBC Driver Installation for UNIX/Linux on an Unsupported Platform 6-2
Configuring a JDBC Data Source (Windows Only) 6-3
JDBC Data Source Configuration Steps 6-3
JDBC Data Source Configuration Procedure 6-3

7 Installing and Configuring OLE DB

Installing OLE DB Provider 7-1
OLE DB Provider Installation Steps 7-1
OLE DB Provider Installation Procedure 7-2
Configuring an OLE DB Data Source 7-2
OLE DB Data Source Configuration Validation Steps 7-3
OLE DB Data Source Configuration Validation Procedure 7-3

Appendix A: ODBC Configuration Files

Netezza ODBC Driver A-1

ODBC Driver Search Behavior A-1
ODBC Driver Considerations A-2
unixODBC Driver Manager A-2
unixODBC Driver Manager Search Behavior A-2
Compatibility. A-2
DataDirect Driver Manager A-3
DataDirect Driver Manager Search Behavior A-3
Incompatibilities A-3
Investigating Client System Configuration A-4

Appendix B: Troubleshooting

ODBC Debug Logging and Driver Manager Tracing B-1
ODBC Debug Logging. B-1
ODBC Driver Manager Tracing. B-2
JDBC Debug Logging B-3
loglevel. B-3
loggertype B-4
logdirpath B-4
Connection String Examples B-4
OLE DB Debug Logging B-4

Index

List of Tables

Table 2-1: Netezza Supported Platforms 2-1
Table 2-2: OLE DB Windows Support 2-2
Table 2-3: Implementation, Netezza Driver/Provider, and Software Compatibility . 2-2
Table 2-4: Netezza ODBC Driver Version Support 2-3
Table 2-5: Data Source Support Changes 2-4
Table 3-1: Minimum Space Requirements for Clients 3-1
Table 3-2: Environment Variables 3-2
Table 4-1: ODBC Driver Properties 4-2
Table B-1: Log Levels B-3

List of Figures

Figure 1-1: ODBC Configuration 1-2
Figure 1-2: JDBC Configuration. 1-4
Figure 3-1: Add or Remove Programs Window. 3-5
Figure 3-2: Program Entry. 3-5
Figure 5-1: ODBC Data Source Administrator Window. 5-4
Figure 5-2: Create New Data Source Window. 5-5
Figure 5-3: System DSN Tab. 5-5
Figure 5-4: ODBC Driver Setup Window DSN Options. 5-6
Figure 5-5: System DSN Advanced Configuration Options. 5-7
Figure 5-6: SSL DSN Options. 5-8
Figure 5-7: Driver Options. 5-9
Figure 6-1: List of Data Sources screen. 6-4
Figure 6-2: Datasource Configuration Connection Options. 6-4
Figure 6-3: Datasource Configuration Advanced Options. 6-5
Figure 6-4: Datasource Configuration SSL Options. 6-5
Figure 6-5: List of Data Sources. 6-6
Figure 7-1: OLE DB Provider: Selection Tab. 7-4
Figure 7-2: OLE DB Provider: Connection Tab. 7-5
Figure 7-3: OLE DB Provider: Connection Succeeded Message. 7-5
Figure 7-4: OLE DB Provider: Advanced Tab. 7-6
Figure B-1: Enabling Debug Logging B-2

Preface

This document describes the Netezza appliance implementations of International standards for a *Structured Query Language Call Level Interface (SQL CLI)*.

This *Netezza Performance Server ODBC, JDBC and OLE DB Installation and Configuration Guide* brings together all the information you need to install and configure the Netezza ODBC drivers, JDBC drivers, OLE DB, and Netezza appliance data sources.

Audience

This guide is written for Netezza appliance installers and troubleshooting staff, database administrators, database programmers, and production analysts.

About this Guide

The purpose of this guide is to document the roles that the SQL CLI, ODBC, JDBC and OLE DB play in the Netezza appliance database environment. The guide supplies background information and presents detailed procedures for installing and configuring driver managers, Netezza ODBC and JDBC drivers, the OLE DB provider, and data sources on all Netezza-supported platforms.

Topic Overview

The following provides a brief overview of the contents of each chapter in this guide.

- **Chapter 1. Introduction**. Describes the changes to this manual version, provides background on ODBC, JDBC, and OLE DB, describes the architecture of ODBC and JDBC, and lists the process steps for installing ODBC, JDBC, and OLE DB.
- **Chapter 2. Supported Platforms and Release Compatibility.** Lists the platforms that the Netezza appliance supports and which software releases are compatible.
- **Chapter 3. Installing and Uninstalling Client Tools.** Details the steps and procedures for installing and uninstalling the Netezza appliance client tools.
- **Chapter 4. ODBC Drivers and Driver Managers.** Discusses the functions of Netezza ODBC and JDBC drivers and ODBC driver managers.
- **Chapter 5. Installing and Configuring ODBC.** Describes procedures for installing and configuring ODBC and configuring Netezza appliance data sources.
- **Chapter 6. Installing and Configuring JDBC.** Describes procedures for installing and configuring JDBC and configuring Netezza appliance data sources.
- **Chapter 7. Installing and Configuring OLE DB.** Describes procedures for installing and configuring OLE DB and configuring Netezza appliance data sources.
- **Appendix A. ODBC Configuration Files.** Describes advanced considerations concerning the names and locations of the ODBC configuration files.
- **Appendix B. Troubleshooting.** Describes the methods for enabling debug logging and tracing, ways of tracking activity to aid in troubleshooting.

Symbols and Conventions

This guide uses the following typographical conventions:

- Italics for emphasis on terms and user-defined values such as user input
- Upper case for SQL commands; for example INSERT, DELETE
- Bold for command line input; for example, **nzsystem stop**

If You Need Help

If you are having trouble using the Netezza appliance, you should:

1. Retry the action, carefully following the instructions given for that task in the documentation.
2. Go to the Netezza Knowledge Base at https://knowledge.netezza.com. Enter your support username and password. You can search the knowledge base or the latest updates to the product documentation. Click Netezza HelpDesk to submit a support request.
3. If you are unable to access the Netezza Knowledge Base, you can also contact Netezza Support at the following telephone numbers:
   - North American Toll-Free: +1.877.810.4441
   - United Kingdom Free-Phone: +0.800.032.8382
   - International Direct: +1.508.620.2281

For a description of the Netezza Support plans, refer to http://www.netezza.com/support. Refer to your Netezza maintenance agreement for details about your support plan choices and coverage.

Netezza Welcomes Your Comments

Let us know what you like or dislike about our manuals. To help us with future versions of our manuals, we want to know about any corrections or clarifications that you would find useful.

Include the following information:

- The name and version of the manual that you are using
- Any comments that you have about the manual
- Your name, address, and phone number

Send us an e-mail message at the following address: doc@netezza.com

The doc alias is reserved exclusively for reporting errors and omissions in our documentation.

We appreciate your suggestions.

Learning More

Find detailed specifications for the Microsoft ODBC implementation of the SQL CLI standards at the following:

http://msdn.microsoft.com/en-us/library

Find more information on the Oracle Sun Developer Network JDBC implementation of the SQL CLI at the following:

http://java.sun.com/reference/api/index.html

Find more information on the Microsoft OLE DB implementation at the following:

http://msdn.microsoft.com/en-us/library

CHAPTER 1

Introduction

What's in this chapter

- Background
- Architecture
- Process
- Installation Files Access

This chapter provides general information about ODBC, JDBC, and OLE DB.

Background

The International standard for *SQL* defines a standard high-level language for accessing and manipulating the data stored in a relational database. The International standard for the *SQL CLI* defines an *application programming interface (API)* that programs use to issue SQL statements in a standardized manner through a series of well-defined API functions.

The term *Open Database Connectivity (ODBC)* is a more commonly used term than SQL CLI, and in many contexts, the two terms are used synonymously. ODBC is actually a Microsoft specification for an API that constitutes an implementation of the International standards for the SQL CLI. ODBC is a superset of the SQL API, since it defines many functions that the International standards for the SQL API do not define. However, the SQL CLI and the Microsoft ODBC specification are closely aligned, and the two terms are used interchangeably especially in the C and C++ programming environments. Unless otherwise noted, the terms SQL CLI and ODBC mean the same thing in this document.

Sun Microsystems developed a specification, called *JDBC*, that defines an API for the Java programming environment that Java programmers use to access relational databases and issue SQL statements. The JDBC specification defines many of the same types of functions for Java programmers that ODBC and the SQL CLI define for C and C++ programmers. It is possible for Java programs to directly invoke the API functions defined by ODBC and the SQL API. However, the nature of the Java programming environment makes this cumbersome. JDBC often provides a better solution than ODBC for the object-oriented Java programming environment.

Like ODBC, JDBC is an implementation of the International standards for the SQL CLI. The JDBC implementation consists of a set of Java interfaces, classes, and methods that a Java programmer can use to do the same types of functions a C or C++ programmer uses when invoking ODBC function calls. Also like ODBC, JDBC constitutes a superset of the SQL CLI, since it defines many functions that the International standards for the SQL CLI do not define.

OLE DB was designed as a higher-level replacement for, and successor to, ODBC, to support a wider variety of non-relational databases that do not necessarily implement SQL (for example, object databases and spreadsheets). OLE has a client-provider model, where clients request access to data, and providers are the software component interface allowing access to the data.

Architecture

This section describes the architectural components of the various implementations.

ODBC

The ODBC implementation is a client-server architecture, with the following major components:

- Application. The application is responsible for interacting with the user and for invoking ODBC API functions that submit SQL statements for processing by a file system or a database management system.
- Driver Manager. The Driver Manager accepts ODBC API functions from the application and passes them to an ODBC driver for processing. It also accepts results from the ODBC driver and passes them to the Application.
- ODBC Driver. The ODBC Driver processes the ODBC API functions it receives from the Driver Manager, interacts with the file system or database management system to access the required data, and returns results to the ODBC Driver Manager.
- Data Source. A Data Source contains the sets of data the ODBC Driver accesses, along with all the environments associated with the data. These environments might include file access or database access software, an operating system, and a networking platform.

To use ODBC to access a Netezza appliance, you need at least a two-tier client-server configuration, as shown in the example in Figure 1-1.

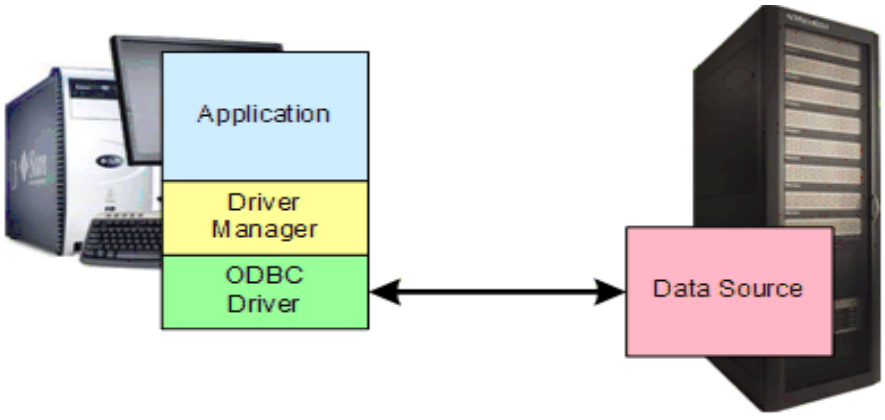

Figure 1-1: ODBC Configuration

In a Netezza appliance two-tier client-server system, the data source is stored on the Netezza appliance, which handles all database access processing. The ODBC driver on the client system uses a network connection to a Postgres process running on the Netezza appliance host, which coordinates SQL statement processing functions.

JDBC

The JDBC architecture consists of four major components:

- Java Application. The application, written in the Java programming language, is responsible for interacting with the user and for invoking JDBC API functions. These API functions submit SQL statements for processing by a file system or a database management system. The Java application uses interfaces, classes, and methods, defined by the Java API, to connect to and access a data source.
- Java Virtual Machine. When a Java application is compiled, the Java object program consists of byte codes. The Java virtual machine is responsible for converting the Java byte codes into instructions capable of being executed on the machine on which the application is running. The Java virtual machine functions as an intermediary between the Java object program and the machine on which the application is running. There is a different Java virtual machine implementation for each platform. The Java virtual machine makes it possible for the same Java program to be run, without changing or recompiling, on any machine for which a Java virtual machine has been implemented.
- JDBC Driver. The JDBC driver processes the JDBC API interfaces, classes, and methods in the Java application. The Java application uses the JDBC driver to interact with the file system or database management system to access the required data. It also returns results to the Java Virtual Machine, which, in turn, returns those results to the Java Application. There is a different JDBC driver for each vendor's DBMS or data source type. Netezza provides a JDBC driver for Version 3.0 of the Oracle JDBC API specification. The driver is a Type 4, pure-Java driver that Netezza has tested for conformance with the Oracle JDBC specifications on Windows, Linux, and UNIX platforms. The Netezza JDBC driver supports all NZSQL data types, includes password encryption, and provides for connection options for host, port, and database. Starting with Netezza appliance Release 4.6, the JDBC driver requires Java release 1.5 and later.
- Data Source. A Data Source contains the sets of data the JDBC driver accesses, along with all the environments associated with the data. These environments might include file access or database access software, an operating system, and a networking platform.

To use JDBC to access a Netezza appliance, you need at least a two-tier client-server configuration, as shown in the example in Figure 1-2.

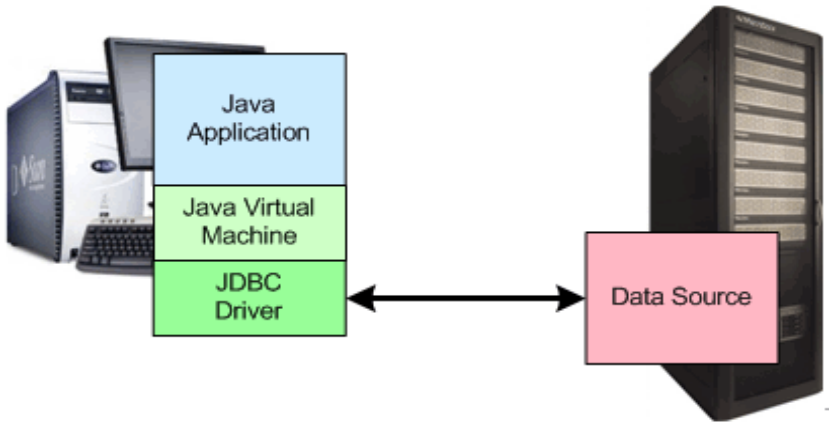

Figure 1-2: JDBC Configuration.

In a Netezza appliance two-tier client-server system, the data source is stored on the Netezza appliance, which handles all database access processing. The JDBC driver on the client system uses a network connection to a Postgres process running on the Netezza appliance host, which coordinates SQL statement processing functions.

Process

This section describes the process steps for ODBC, JDBC, and OLE DB, and where to go for more detail.

ODBC

The process for setting up ODBC on your system is as follows.

1. Install the appropriate Netezza appliance client tools. See the next section "Installation Files Access" and Chapter 3, "Installing and Uninstalling Client Tools."
2. (UNIX only) Download an ODBC driver manager. See Chapter 4, "ODBC Drivers and Driver Managers."
3. Select the ODBC driver to install. See Chapter 4, "ODBC Drivers and Driver Managers."
4. Configure the data source. See Chapter 5, "Installing and Configuring ODBC."
5. Test connectivity. See Chapter 5, "Installing and Configuring ODBC."

JDBC

The process for setting up JDBC on your system is as follows.

1. Install the appropriate Netezza appliance client tools. See the next section "Installation Files Access" and Chapter 3, "Installing and Uninstalling Client Tools."
2. Install the JDBC driver. See Chapter 6, "Installing and Configuring JDBC."
3. Configure the data source. See Chapter 6, "Installing and Configuring JDBC."

OLE DB

The process for setting up OLE DB on your system is as follows.

1. Install the appropriate Netezza appliance client tools. See the next section "Installation Files Access" and Chapter 3, "Installing and Uninstalling Client Tools."
2. Select the OLE DB provider to install. See Chapter 7, "Installing and Configuring OLE DB."
3. Configure the data source. See Chapter 7, "Installing and Configuring OLE DB."

Installation Files Access

You can access the files you need for installation from the Netezza FTP support site, or from the *Netezza Client* CD.

- Netezza FTP support site — navigate here using either the command line method or through a Web browser.
  - To access from a command line prompt, enter the following command:

    `ftp ntzftp.netezza.com`
  - To access from a Web browser, enter one of the following URLs:

    `http://ntzftp.netezza.com`

    `https://ntzftp.netezza.com`

  Note: You must supply your Netezza FTP user account and password to access the server, where you will begin in your home directory location.
- *Netezza Client* CD — Insert the CD in a system drive and use Windows Explorer to open the **`\admin`** folder on the CD.

To access the files, navigate to the Releases/*version* directory, where *version* is the Netezza software version with the tools that you want to download. For example, to navigate to the Netezza Release 6.0 software location, navigate to Releases/6.0. As a best practice, check for a patch subdirectory to see if there is a more recent set of client installation executables.

Select the release whose client tools you want to install, and the page updates to display links to the files. The software version directory contains files such as the following:

- *cli.package.tar — Netezza client installation packages for the supported client operating systems. Note that there are different packages for the different clients.
- datadirect.package.tar.z — DataDirect Driver Manager for ODBC support
- doc.package.tar.z — Netezza documentation for the release
- nzjdbcsetup.exe — JDBC driver for Netezza clients
- nzodbcsetup.exe — ODBC driver for Netezza clients
- nzoledbsetup.exe and nzoledbsetup64.exe — OLE DB provider for Netezza clients (32- and 64-bit installers, respectively)
- nzsetup.exe — NzAdmin client installation kit (Windows clients)
- package.tar.z — Netezza software installation files

If you need one of the .tar files, use the **tar** command to uncompress the package.tar.z file, for example:

```
tar cli.package.tar.z
```

You can download client kits such as ODBC, JDBC, OLE DB, or Netezza client CLIs for your Netezza client users to update to the latest software release. The downloads will vary depending upon the clients that you use, but you should ensure that you and your Netezza users upgrade to the latest clients for best performance.

CHAPTER 2

Support and Release Compatibility

What's in this chapter

- Supported Platforms
- Release Compatibility Matrix
- ODBC Driver Version Support
- ODBC Standards Conformance and API Versions
- JDBC Driver Data Source Support

This chapter shows the supported platforms and compatibility.

Supported Platforms

Netezza supplies the following platforms for ODBC and JDBC:

Table 2-1: Netezza Supported Platforms

Operating System	32-bit	64-bit
Windows		
Windows 2000	X	
Windows 2003	X	X
Windows 2008	X	X
Windows XP	X	X
Windows Vista	X	X
Windows 7	X	X
Linux		
Red Hat LAS Linux 4.0, 5.2, 5.3, 5.5	Intel/AMD	Intel/AMD
SUSE Linux Enterprise Server 8 and 9	Intel/AMD	Intel/AMD

Table 2-1: Netezza Supported Platforms (continued)

Operating System	32-bit	64-bit
UNIX		
Oracle Solaris 8, 9, 10	SPARC	SPARC
Oracle Solaris 10	x86	x86
HP-UX 11i versions 1.6 and 2 (B.11.22 and B.11.23)	Itanium PA-RISC	Itanium PA-RISC
IBM AIX 5.1, 5.2, 5.3, 6.1 with 5.0.2.1 C++ runtime libraries	PowerPC	PowerPC

The OLE DB provider is supported on the Windows clients shown in Table 2-2:

Table 2-2: OLE DB Windows Support

Operating System	32-bit	64-bit
Windows 2003 Server	X	X
Windows 2008 Server	X	X
Windows XP	X	X
Windows Vista Business	X	X
Windows 7	X	X

Release Compatibility Matrix

Table 2-3 shows the previous version compatibility between the implementation, the Netezza driver/provider versions, and Netezza appliance software versions.

Note: The software and driver/provider versions shown are baseline versions. Always check the Netezza site for the latest version to ensure up-to-date capability.

Table 2-3: Implementation, Netezza Driver/Provider, and Software Compatibility

Implementation	Netezza Driver/ Provider	Netezza Appliance Software
ODBC	4.5.x 4.6.x 5.0.x 6.0.x	3.1.x, 4.0.x, 4.5.x, 4.6.x, 5.0.x 4.0.x, 4.5.x, 4.6.x, 5.0.x, 6.0.x 4.5.x, 4.6.x, 5.0.x, 6.0.x 4.6.x, 5.0.x, 6.0.x
JDBC	4.5.x 4.6.x 5.0.x 6.0.x	3.1.x, 4.0.x, 4.5.x, 4.6.x, 5.0.x 4.0.x, 4.5.x, 4.6.x, 5.0.x, 6.0.x 4.5.x, 4.6.x, 5.0.x, 6.0.x 4.6.x, 5.0.x, 6.0.x

Table 2-3: Implementation, Netezza Driver/Provider, and Software Compatibility

Implementation	Netezza Driver/ Provider	Netezza Appliance Software
OLE DB	4.5.x 4.6.x 5.0.x 6.0.x	4.0.x, 4.5.x, 4.6.x, 5.0.x 4.0.x, 4.5.x, 4.6.x, 5.0.x, 6.0.x 4.5.x, 4.6.x, 5.0.x, 6.0.x 4.6.x, 5.0.x, 6.0.x

Note: Different driver/provider and software versions may have different available functions, so for full functionality, you may need to change your driver/provider or software version.

ODBC Driver Version Support

Table 2-4 shows the Netezza support for the Microsoft ODBC driver versions.

Table 2-4: Netezza ODBC Driver Version Support

Microsoft ODBC Version	Windows Platforms	UNIX/Linux Platforms
ODBC 3.0	Supported	Supported
ODBC 3.5	Supported	Supported

A distinguishing feature of the Microsoft ODBC Release 3.x specifications is that they are closely aligned with the X/Open and ANSI/ISO/IEC International standards for the SQL CLI.

When you install a Netezza ODBC driver in a client computer system, you must choose one of the available drivers. You cannot install multiple Netezza ODBC drivers in the same client system.

ODBC Standards Conformance and API Versions

The Netezza ODBC drivers conform to the Microsoft specification for the Open Database Connectivity (ODBC) API and the following International standards for the SQL Call Level Interface (SQL CLI):

- The X/Open CAE Specification “Data Management: SQL Call-Level Interface (CLI)”
- ISO/IEC 9075-3:1995 (E) Call Level Interface (SQL CLI)

The X/Open and ISO/IEC standards for the SQL CLI are published in separate documents, but the two standards are closely aligned.

JDBC Driver Data Source Support

Up to 4.5.x, JDBC driver implementations for data sources had some limitations, such as not allowing users to set different connection properties, such as: securityLevel, caCertFile, loggerType, etc. In 4.6 these data sources are removed and improved data sources are added, as shown in the following:

Table 2-5: Data Source Support Changes

JDBC Interface	Former Implementation	New Implementation	Remarks
javax.sql.DataSource	org.netezza.jdbc2.optional.SimpleDataSource	org.netezza.datasource.NzDatasource	Basic DataSource implementation.
javax.sql.DataSource	org.netezza.jdbc2.optional.PoolingDataSource	org.netezza.datasource.NzConnectionPool	Basic DataSource implementation with connection pooling implemented inside it. Use the connections provided by this data source directly as normal connections. The Data Source internally handles the pooling. Do not use this data source if your server/ middleware vendor provides a connection pooling implementation. This data source is provided as a convenience only.
javax.sql.ConnectionPoolDataSource	org.netezza.jdbc2.optional.ConnectionPool	org.netezza.datasource.NzConnectionPoolDatasource	Basic ConnectionPoolDataSource implementation.

CHAPTER 3

Installing and Uninstalling Client Tools

What's in this chapter

- Windows Client Tools Installation
- UNIX/Linux Client Tools Installation
- Uninstalling Client Tools for Windows
- Uninstalling Client Tools for UNIX/Linux

This chapter provides information about installing the Netezza appliance client tools on your system. These tools help the ODBC drivers, JDBC drivers, and OLE DB Provider that Netezza supplies connect to a Netezza appliance from a client system.

The first step in preparing a system for ODBC, JDBC, or OLE DB is to install the Netezza appliance client tools on your system. If these are already installed, skip this chapter. After the client tools have been installed, you can install the ODBC drivers, JDBC drivers, or OLE DB Provider.

Windows Client Tools Installation

The installation program for Windows runs a standard InstallShield Wizard to install the Netezza appliance tools. During the process, you can back up and change any selections you made or information you supplied, and you can cancel the installation at any time.

Depending on the selections you make during the installation process, the installation program adds appropriate entries to the Windows registry and installs the following:

- Netezza administration application
- Console program executable files and libraries
- Console program string files
- Online help files
- Documentation files in PDF format

Table 3-1 lists the minimum disk space required to install each client.

Table 3-1: Minimum Space Requirements for Clients

Installing	Disk Space Needed
ODBC	100 MB

Table 3-1: Minimum Space Requirements for Clients

Installing	Disk Space Needed
JDBC	10 MB
OLE DB	100 MB

Table 3-2 describes the environment variables the installation program adds for the Netezza Performance Server console applications.

Table 3-2: Environment Variables

Variable	Operation	Setting
NZ_DIR	set	Netezza Appliance Tools Installation directory (for example C:\Program Files\ Netezza)
PATH	append	<installation directory>\bin

Created Subfolders

The installation program creates the following subfolders within the specified installation location. Do not change the names or locations of these installation subfolders.

- **Bin** — Executables and support files
- **Doc** — Netezza product documentation
- **Help** — Application help files
- **Log** — Empty directory used for application logging
- **Sys** — A Strings subfolder containing loader string files
- **Tmp** — Empty directory used for temporary storage

Client Tools Installation Procedure for Windows

The examples illustrate the procedure on a Windows XP system. The procedure is similar for other Windows systems, but the windows differ slightly.

Access the **`nzsetup.exe`** program. For directions, see “Installation Files Access” on page 1-5. After the **`nzsetup.exe`** program starts, it displays the Welcome window.

1. Click **Next** to begin the software installation process.
2. The License Agreement window is displayed. Read the agreement, and then click **I accept the terms in the license agreement**. You cannot install the tools if you do not accept the license agreement. After you have read and accepted the license agreement, click **Next** to continue.
3. The next window lets you change the installation folder or accept the Netezza default. The Netezza default installation folder is **`Program Files\Netezza`**. Select the default or click **Change** to select a different folder. If the specified folder does not exist on your system, the installation program creates it. After changing the folder or accepting the default, click **Next**.

4. The Setup Type window is displayed. Select from the following installation types:
   - **Typical** — (Recommended for most client systems) Installs the **`nzadmin`** program, the online help files, the documentation, and the console utilities, including the loader.
   - **Minimal** — Installs the **`nzadmin`** program and online help files.
   - **Custom** — Displays a screen that allows you to specify whether you want to install the administration application, the console application, or the documentation.

   The installation program installs all the Netezza client software, help files, and documentation.
5. The Review window is displayed, summarizing your installation choices. To change any of your choices, use the Back button to page through the previous windows. When finished, click **Install**.

   After you click **Install**, the installation program installs the Netezza appliance tools software and places appropriate entries in the Windows registry. Windows displays a series of progress dialog boxes as the installation program does the various steps.

   The installation program adds a Netezza appliance tools folder in the Programs folder on your Start menu. From the Start menu, you can access the Netezza Server Administrator application (**`NzAdmin`**).
6. The Installation Completed window is displayed, indicating that the installation process has completed successfully. Click **Finish** to exit the installation procedure. On some systems, the installation program may prompt you to reboot your system.

   Note: To run the nz CLI commands, launch them from a Windows Command Prompt window.

UNIX/Linux Client Tools Installation

This section describes the process for installing the Netezza aplliance client tools for a UNIX/Linux system.

Created Subdirectories

The client tools installation program for a UNIX/Linux system creates the following subdirectories in the **`nz`** directory:

- **bin** — Contains the Netezza CLI command software.
- **doc** — Contains the user documentation.
- **lib** — Contains ODBC, JDBC, and OLE DB software.
- **license** — Contains the Netezza end-user software license agreement.
- **sys/strings** — Contains strings for use by the Netezza CLI commands.
- **tmp** — An empty directory used to contain the temporary files Netezza software creates.

Client Tools Installation Procedure for UNIX/Linux

This section describes the steps for installing the Netezza client tools on a UNIX/Linux client system.

1. Run the client program for the type of system you are installing on. For example, for a 64-bit Linux system, you would select **linux64cli.package.tar**. For directions, see "Installation Files Access" on page 1-5. Make sure to mount it, if it is not automatically mounted on your system.
2. The installation default causes the installation program to create the installation directory in a directory relative to the root of the file system. So to use the default installation directory, log in as the superuser to do the installation. Only the superuser has write access to the root of the file system. Installing the Netezza client tools as the superuser allows all users of the client system to access the Netezza client tools, data files, ODBC drivers, and JDBC driver.

   Note: In some cases, you can log in as an ordinary user to install the Netezza client tools. To do so, create the installation directory in a directory for which you have write access. The procedures in this section assume you can log in as the superuser when installing software. If not, consult your technical support department to see what changes to make.
3. Change to the directory corresponding to the platform to install. For example, to change to the directory for Linux, type the following command:

   ```
   cd /mnt/cdrom/linux
   ```
4. Run the setup command, which checks the client OS to identify the correct client files and launches the unpack command, or navigate to the correct OS directory and then run the unpack command as follows:

   ```
   ./unpack
   ```
5. The installation program allows you to make the following installation choices:
   - View the **readme** file (optional).
   - Accept the default installation directory or change it. Netezza recommends you accept the default installation directory.
   - Select to install the ODBC driver, the JDBC driver, or the OLE DB provider. For more information on each option, see the appropriate chapter.
6. If you accepted the default for the installation directory, the installation program creates the **/usr/local/nz** directory and copies the installation files into subdirectories within the **nz** directory. If you changed the default installation directory, the installation program creates the **nz** directory in the directory you specified.

   Note: If you direct the installation program to create the installation directory in a directory to which you do not have write access, the program fails and issues a message that it was unable to create the **nz** directory. This failure most often occurs if you log in as an ordinary user for the installation and do not change the installation directory appropriately.

Uninstalling Client Tools for Windows

This sections shows an example of uninstalling on a Windows XP system.

To remove the Netezza appliance client tools for Windows for the ODBC driver, the JDBC driver, and/or the OLE DB provider, use the standard Add or Remove Programs control panel interface. Click **Start > Control Panel > Add/Remove Programs** and select the Netezza entries on the list of installed programs.

Figure 3-1 shows the Add or Remove Programs Control Panel window.

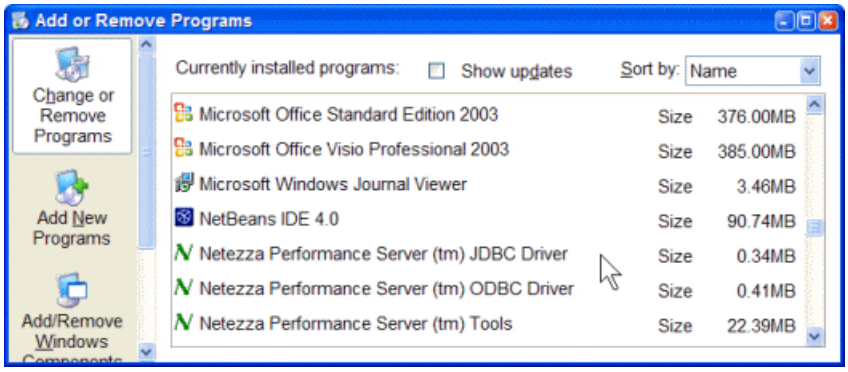
Add or Remove Programs
Change or Remove Programs
Add New Programs
Add/Remove Windows
Currently installed programs:
Show updates
Sort by: Name
Microsoft Office Standard Edition 2003 Size 376.00MB
Microsoft Office Visio Professional 2003 Size 385.00MB
Microsoft Windows Journal Viewer Size 3.46MB
NetBeans IDE 4.0 Size 90.74MB
Netezza Performance Server (tm) JDBC Driver Size 0.34MB
Netezza Performance Server (tm) ODBC Driver Size 0.41MB
Netezza Performance Server (tm) Tools Size 22.39MB

Figure 3-1: Add or Remove Programs Window.

Select the entry in the Add/Remove list to remove. Windows expands the entry and displays the Change and Remove buttons, as shown in Figure 3-2.

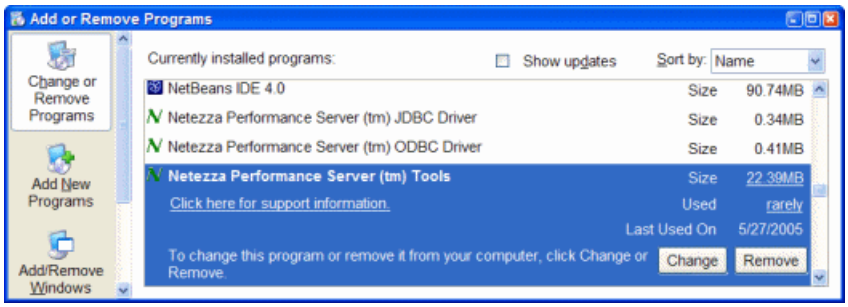
Add or Remove Programs
Change or Remove Programs
Add New Programs
Add/Remove Windows
Currently installed programs:
Show updates
Sort by: Name
NetBeans IDE 4.0 Size 90.74MB
Netezza Performance Server (tm) JDBC Driver Size 0.34MB
Netezza Performance Server (tm) ODBC Driver Size 0.41MB
Netezza Performance Server (tm) Tools Size 22.39MB
Click here for support information. Used rarely
Last Used On 5/27/2005
To change this program or remove it from your computer, click Change or Remove.
Change
Remove

Figure 3-2: Program Entry.

The Windows 2000, 2003, Windows Vista, and Windows XP Add/Remove Programs control panel may give you the option of removing, repairing, or changing the installation. You may also be able to access support information, such as technical support links and phone numbers, product update information, and information about the company.

The uninstall procedure removes all folders, files, menu commands, and environment variables. Registry entries created by other Netezza applications are not removed.

Uninstalling Client Tools for UNIX/Linux

There is no explicit UNIX/Linux “uninstall” program for the Netezza appliance client software. To remove the Netezza appliance client software from your system, log on as the superuser and use the **`rm`** command to delete the installation directory and all of its subdirectories. The default installation directory is **`/usr/local/nz`**, unless you changed it when you installed the Netezza appliance client tools software and ODBC/JDBC drivers.

The following is an example of commands to delete the software:

```
cd /usr/local
rm -f -r nz
```

The above commands remove the **nz** directory and all of its subdirectories and files from the **/usr/local** directory.

CHAPTER 4

ODBC Drivers and Driver Managers

What's in this chapter

- ODBC Drivers
- ODBC Driver Managers

This chapter explains ODBC drivers and driver managers.

You can have only one ODBC driver installed on your system at a time. If you need a different driver, remove the installed driver and install the different one. For more information on this procedure, see Chapter 3, "Installing and Uninstalling Client Tools." For driver installation and configuration detail, see Chapter 5, "Installing and Configuring ODBC."

ODBC Drivers

An ODBC driver uses a layer of code that translates the ODBC standard API calls the application issues into vendor-specific functions. Netezza supplies different versions of the ODBC driver for each supported platform, each vendor's DBMS or data source type, each of the major ODBC versions, and for each major release level of the Netezza appliance.

ODBC drivers expose the capabilities of the underlying data source, and each ODBC driver is designed for a particular database management system. For example, an ODBC driver designed to access a different vendor database cannot be used to access a Netezza database.

ODBC Driver Functions

ODBC drivers do a variety of functions, including the following:

- Connecting to and disconnecting from data sources.
- Performing error checking functions.
- Submitting SQL statements to the database management system software for execution against a particular data source.
- Converting ODBC-standard SQL statements, if necessary, to DBMS-specific SQL statements.
- Sending data to and retrieving data from a data source.
- Performing data conversion functions, as specified by the application.
- Returning information about the success or failure of an ODBC function.

ODBC Driver Properties

This section describes the meaning and purpose of each ODBC driver property. Parameters define the properties of Netezza ODBC drivers, configured either through Windows or a UNIX/Linux configuration file. Use this information to select appropriate values for each driver property during the driver configuration process.

Table 4-1: ODBC Driver Properties

Property	Setting
DebugLogging	This boolean property activates debug logs. By default, logging is disabled. To enable logging, select the property on the Windows dialog, or for UNIX, specify a boolean such as 1 or true. For more information, see Appendix B, "Troubleshooting."
Log File Path	The location of the debug log files. The default is /tmp on UNIX/Linux systems or c:\ on Windows systems.
Prefetch Count	A numeric value that sets the number of rows the driver will fetch at a time from a Netezza database. The default is 256 rows. To tune your application, set a value that optimizes network use versus memory use. Realize that the higher this value, the more memory will be required to hold these rows.
Socket Buffer Size	A numeric value that specifies the size of the communications buffer in bytes. The range is 1K to 32K. The default is 8192.
Character Translation Option	The Netezza appliance uses the Latin9 character encoding for char and varchar types. The character encoding for many Windows systems is similar, but not identical. If your database includes characters that use only the basic subset of letters (a-z or A-Z), numbers (0-9), or punctuation characters, select the Optimize for ASCII character set option for the driver on Windows, which improves query performance. If your data includes special characters such as the Euro symbol or others, de-select the option so that the characters convert correctly.
Unicode Translation Option	For UNIX/Linux drivers, specifies the unicode encoding value. Valid values are utf8, utf16, and utf32. The default is utf8.

ODBC Driver Managers

The ODBC driver manager serves as an intermediary between an application and an ODBC driver. It provides a single place for an application to invoke most ODBC API functions. In most DBMS application environments, including the Netezza appliance environment, the application is linked to the driver manager and is not linked directly to a specific ODBC driver.

The driver manager is the ODBC component that manages communication between applications and ODBC drivers. In most cases, ODBC applications issue ODBC API calls to the driver manager, and the driver manager, in turn, passes those ODBC API function calls to the appropriate ODBC driver. The ODBC driver then accesses the underlying database management system software.

For performance reasons, depending on the database management system software used, some API functions may bypass the driver manager and access the ODBC driver directly.

ODBC Driver Manager Functions

The driver manager handles a number of functions that all ODBC applications need, regardless of the particular ODBC driver they use or the data source they access. The functions of the driver manager include the following:

- Loading into memory the ODBC driver required by the application and the data source it is accessing.
- Unloading from memory ODBC drivers when they are no longer needed.
- Passing ODBC API function calls from the application to an appropriate ODBC driver.
- Performing any required ODBC API function call conversions. For example, if an ODBC application issues ODBC 2.5 function calls and is using an ODBC 3.0 driver, the driver manager converts ODBC 2.5 function calls to their ODBC 3.0 equivalents when necessary.
- Performing error checking and validation functions on ODBC API function calls before passing them to the ODBC driver.
- Performing some character set conversions. For example, if the application uses an 8-bit ASCII character set, and the ODBC driver and DBMS work with character data in 16-bit Unicode format, the driver manager does the necessary character conversions.

Note: The driver manager component is used only with ODBC. There is no separate driver manager component in the JDBC architecture. With JDBC, all driver manager functions are performed in the Java application environment.

Windows Driver Managers

Microsoft provides a driver manager for the Windows environment. The ODBC driver manager is part of the *Microsoft Data Access Component (MDAC)* and, as such, is part of the Windows operating system. Therefore, you do not have to install a driver manager in the Windows environment to use the Netezza ODBC drivers.

Linux/UNIX Driver Managers

To use the Netezza ODBC drivers in a Linux/UNIX environment, you must install an appropriate driver manager. You can get driver manager components from a number of sources.

- Open Source — There are a number of driver managers available on the Web from open source software projects. Netezza supports the *unixODBC* driver manager component for Linux/UNIX platforms. You can find information about unixODBC at the following:

  `http://www.unixodbc.org`

- Commercial — A number of commercial software vendors develop and market ODBC components, including driver managers. Netezza has licensed and provides support for the DataDirect driver manager from *DataDirect Technologies*. It is included with your system on it's own CD. The DataDirect driver manager component comes with some

third-party ETL and BI software tools. It is not necessary to install the Netezza-licensed version of the DataDirect driver manager to access the Netezza ODBC drivers. You can find information on the DataDirect line of ODBC products at the following:

```
http://www.datadirect.com
```

unixODBC Driver Manager Installation

This section provides instructions for installing the open source unixODBC driver manager. If your installation uses some other driver manager component, skip this section. The latest version of the unixODBC driver manager software can be downloaded from the following:

```
http://www.unixodbc.org
```

The following procedure example is based on version 2.2.12, and assumes the name of the downloaded file is **`unixODBC-2.2.12.tar.gz`**. The file name might be different if you download a later release. Match the file and directory names in the procedure to correspond to the name of the file you download. Consult the README and INSTALL files contained in the downloaded package to determine whether the procedure needs to be modified.

unixODBC Driver Manager Installation Steps

The installation procedure for the unixODBC driver manager is the same for all UNIX and Linux systems. To install and configure release 2.2.12 of the unixODBC driver manager on a UNIX/Linux system, do the following steps:

1. Login as superuser.
2. Download the unixODBC driver manager software.
3. Uncompress the downloaded file.
4. Extract the contents of the tar file.
5. Change directory to the unixODBC extraction directory.
6. Configure the unixODBC software.
7. Compile the unixODBC package.
8. Install the programs, data files, and documentation.
9. Add subdirectories to paths.
10. Delete or move the installation file, if desired.

unixODBC Driver Manager Installation Procedure

The following sections describe each of the steps in the procedure for installing the unixODBC driver manager on a UNIX/Linux system.

1. Since the unixODBC installation procedure requires write access to the root of the file system to create installation directories, log in as the superuser to access the system during the installation process to use the default installation directories.
2. Download the driver manager file from the **`www.unixodbc.org`** Web site.
3. Uncompress the downloaded file by issuing the following command:

```
gzip -d unixODBC-2.2.12.tar.gz
```

4. Run the following command to extract the contents of the **unixODBC-2.2.12.tar** file:

   ```
   tar -xvf unixODBC-2.2.12.tar
   ```

5. Run the following command to change to the subdirectory created when the **unixODBC-2.2.12.tar** file was extracted:

   ```
   cd unixODBC-2.2.12
   ```

6. Run the following command to configure the software and set the installation directory to **/usr/local/unixODBC**:

   ```
   ./configure --prefix=/usr/local/unixODBC
   ```

   Note: To configure without the GUI, run the following command:

   ```
   ./configure --prefix=/usr/local/unixODBC --enable-gui=no
   ```

7. Run the following command to compile the package (note that this step may take a few minutes to complete):

   ```
   make
   ```

8. Run the following command to install the programs, data files, and documentation:

   ```
   make install
   ```

   This step places the **bin**, **etc**, **lib**, and **include** subdirectories in the **/usr/local/unixODBC** directory. These subdirectories contain the programs, data files, and documentation for the unixODBC driver manager package.

9. Run the appropriate commands to add **bin** and **lib** subdirectories to the environment:

   For AIX:

   ```
   export PATH=$PATH:/usr/local/unixodbc/bin
   export LIBPATH=$LIBPATH:/usr/local/unixodbc/lib
   ```

   For HP-UX:

   ```
   export PATH=$PATH:/usr/local/unixodbc/bin
   export SHLIB_PATH=$SHLIB_PATH:/usr/local/unixodbc/lib
   ```

   For Linux, Solaris:

   ```
   export PATH=$PATH:/usr/local/unixodbc/bin
   export LD_LIBRARY_PATH=$LD_LIBRARY_PATH:/usr/local/unixodbc/lib
   ```

10. If desired, delete the installation **tar** file from the file system:

    ```
    rm unixODBC-2.2.12.tar
    ```

    If preferred, you could move the installation **tar** file to somewhere else in the file system for safekeeping.

DataDirect Driver Manager Installation

This section provides instructions for installing the Netezza-branded driver manager from DataDirect Technologies. If your installation uses some other driver manager component, skip this section. The following installation and configuration procedure is based on version 5.1. Consult the README and INSTALL files contained in the platform-specific directories to determine whether the procedure needs to be modified.

Assumptions

The installation procedure for the DataDirect driver manager is the same for all supported Linux and UNIX systems. However, the exact file names for the manager are platform-specific.

- The installation directory is **/opt**. If you choose another directory, modify the following files:
  - **odbc.sh** (if using sh or bash)
  - **odbc.csh** (if using csh)
  - **odbc.ini**
- The **odbc.ini** and **odbcinst.ini** files are located in **/opt/odbc32v51** (32 bit) and **/opt/odbc64v51** (64 bit). If you choose another directory, modify the following files:
  - **odbc.sh** (if using sh or bash)
  - **odbc.csh** (if using csh)
  - **odbc.ini**
  - **odbcinst.ini**
- The Netezza ODBC driver is located in **/usr/local/nz/lib**. Verify that the driver entry points to the proper file name. If you choose another directory, modify the following files:
  - **odbc.ini**
  - **odbcinst.ini**

DataDirect Driver Manager Installation

To install and configure release 5.1 of the DataDirect driver manager on a Linux/UNIX system, do the following:

1. Login as the superuser.
2. Copy the appropriate driver manager file to the proper installation directory.
3. Extract the contents of the driver manager file from the chosen installation directory. For example, to extract the AIX 32-bit driver manager, enter the following:

   ```
   tar -xvf aix32_dd_dm.tar
   ```

4. Change to the extraction directory. For example, to change to the 32-bit extraction directory, enter the following:

   ```
   cd odbc32v51
   ```

5. Based on the assumptions above, make the required file changes.
6. Edit odbc.ini and specify the servername, database, login, and password.
7. Execute the appropriate environment file:

   ```
   . ./odbc.sh
   ```

   or

   ```
   source odbc.csh
   ```

8. Change directory to example:

```
cd example
```

9. Run the example program (**example**), specifying a DSN (NZSQL is the default), login, and password. You should see the following output:

```
./example DataDirect Technologies, Inc. ODBC Example Application.
Enter the data source name : NZSQL
Enter the user name        : admin
Enter the password         : password
Enter SQL statements (Press ENTER to QUIT)
SQL>
```

If an error occurs, contact Netezza Customer Support.

CHAPTER 5
Installing and Configuring ODBC

What's in this chapter

- Installing and Configuring ODBC for Windows
- Configuring ODBC for UNIX/Linux

This chapter describes the procedures used to install and configure ODBC drivers and data sources for Windows or UNIX/Linux systems.

You can have only one ODBC driver installed on your system at any given time. If you need a different driver, you will need to remove the installed driver and install the different one. For more information on removing the driver, see Chapter 3, "Installing and Uninstalling Client Tools."

Installing and Configuring ODBC for Windows

This section describes procedure for installing a Netezza ODBC driver on a Windows client system, configuring the driver and data sources, and testing for connectivity to Netezza appliance data sources. If you have already installed a Netezza ODBC driver on your computer system, go to "ODBC Driver Data Source Configuration for Windows" on page 5-3.

The **`nzodbcsetup.exe`** file contains the Netezza ODBC drivers and the ODBC driver installation program. The installation program installs the Netezza ODBC libraries on your system, creates a Netezza SQL system data source entry (NZSQL) with appropriate default values, and adds the appropriate entries to the Windows registry.

You can install the ODBC driver from the command line, or by double-clicking the ODBC installation program, which runs the interactive GUI. The command line option can run the installation via the GUI or in silent mode, useful if you use a software management program to install the ODBC driver on your client systems.

ODBC Driver Installation for Windows via Command Line

Use one of the following options to install from the command line.

Note: Run the nzodbcsetup executable file in the appropriate directory on your system. For directions, see "Installation Files Access" on page 1-5. The following examples use the C drive, however you can install wherever appropriate.

- Full install using the Graphical User Interface (GUI)

  ```
  c:\>nzodbcsetup.exe
  ```

- Silent install; Visual Studio 8 (VC8) libraries are not installed and Translation Options are set to the default value

  ```
  c:\>nzodbcsetup.exe -i silent
  ```

- Silent install; Visual Studio 8 (VC8) libraries are installed and Translation Options are set to the default value

  ```
  c:\>nzodbcsetup.exe -i silent -DVC8_LIB_INSTALL=1
  ```

- Silent install; Visual Studio 8 (VC8) libraries are installed and Translation Options are set to **Optimize for ASCII**

  ```
  c:\>nzodbcsetup.exe -i silent -DVC8_LIB_INSTALL=1
  -DTRANSLATION_OPTION_VALUE=1
  ```

ODBC Driver Installation Steps for Windows via the GUI

The following are the steps for installing the ODBC driver with the interactive GUI:

1. Launch ODBC Driver Installation Program.
2. Select the language for the installation program.
3. Review the Introduction window.
4. Select the Driver Bit Size (if necessary).
5. Select the Character Translation Options for Special Character Support.
6. Install the Visual Studio 8 libraries.
7. Review the installation options.
8. Finish the installation.

ODBC Driver Installation Procedure for Windows via the GUI

This section describes each of the steps to install a Netezza ODBC driver for Windows using the GUI. The examples are for a Windows XP system for version 5.0, but the procedure is the same on other Windows systems.

Complete the following steps to install the ODBC driver.

1. Start the installation program by double-clicking the **nzodbcsetup.exe** program. For directions, see “Installation Files Access” on page 1-5.
2. A language option window is displayed, which controls the language for the installation program. Accept the default of English and click **OK**.
3. The Introduction window is displayed. Review the information and click **Next**.
4. On 64-bit systems, you are given the option of selecting the bit size of the driver to install. Select 32-bit, 64-bit, or both, and click **Next**. On 32-bit systems, this window is not displayed.
5. The Character Translation Option window is displayed. Specify whether you will use only the basic subset of characters, such as letters (a-z or A-Z), numbers (0-9), or punctuation marks in the CHAR and VARCHAR datatypes. If you use only these characters, select the **Optimize for ASCII character set** option to improve data transfer performance. However, if you use characters such as the Euro symbol or other charac-

ters that are outside the basic set, do not select the optimize option. This causes the characters that you enter to be converted to the proper encodings so that they appear correctly in your query results. Click **Next** after selecting.

6. The ODBC driver requires the Visual Studio 8 (VC8) Libraries on the client system.
   - If the installation program finds the VC8 libraries, it uses the installed files and proceeds to the Pre-Installation Summary window.
   - If the program cannot find the VC8 files on the system, it displays the Visual Studios 8 Libraries window. Select **Install** and click **Next**. If you can verify that the VC8 libraries are present on the system, even though the installer could not locate them, select **Do not install** and click **Next**.
   - If the driver does not function properly because the libraries are missing or not found, run the driver installation program again and install the VC8 libraries.
7. The Pre-Installation Summary window is displayed, summarizing your installation choices. To page through the previous selections to review selections, use the **Previous** button. When finished, click **Install**.

   After you click **Install**, the installation program does the following processing:
   - Installs the Netezza ODBC Driver. A 32-bit driver on a 32-bit system and a 64-bit driver on a 64-bit system are installed in the **`\WINDOWS\system32`** folder. If installing both 32- and 64-bit drivers on a 64-bit system, the 32-bit driver is installed in the **`\WINDOWS\SysWOW64`** folder**.**
   - Places entries in the Windows registry to add your selected ODBC driver to the driver list.
   - Places entries in the Windows registry to add the NZSQL data source to the data source list.
8. The Install Complete window is displayed. This indicates that the installation process has completed successfully. Click **Done** to close the program.

   Before trying to access an ODBC data source, you must configure it. The next section describes the configuration procedure.

ODBC Driver Data Source Configuration for Windows

After you install a Netezza ODBC driver, the installation program creates an entry for a system data source called NZSQL. You then configure that data source to connect to a Netezza appliance data source. You can also configure a user data source.

- A *System DSN* defines a data source available to all users of the client system.
- A *User DSN* defines a data source available only to the user who defined it.

Get the following configuration information from your Netezza system administrator:

- The name of the system or user database on the Netezza appliance.
- The server host name or IP address of your Netezza appliance.
- A valid user ID for accessing the system or user data source.
- A valid password for accessing the system or user data source.

Data Source Configuration Procedure

The following outlines the procedure to configure a Netezza appliance data source.

Note: This procedure is for configuring a 32-bit DSN on a 32-bit system or a 64-bit DSN on a 64-bit system. 32-bit DSNs do not display in the 64-bit administrator. On a 64-bit system, launch the administrator using the following: **C:\windows\syswow64\odbcad32.exe**

1. Open the ODBC Data Source Administrator Window and select either the System DSN tab or the User DSN tab for the appropriate configuration. Then select either of the following:
   - ▲ To configure an existing DSN, select the Configure button.
   - ▲ To configure a new DSN, select the Add button.
2. Configure the DSN and Driver Options from the Driver Setup Window.
3. Test and verify the ODBC Connection.
4. Close any open configuration windows to finish the Data Source Configuration.

ODBC Data Source Administrator Window The ODBC Data Source Administrator window allows you to make configuration selections.

1. Select **Start > Settings > Control Panel > Administrative Tools > Data Sources (ODBC)**. The ODBC Data Source Administrator window is displayed, as shown in Figure 5-1:

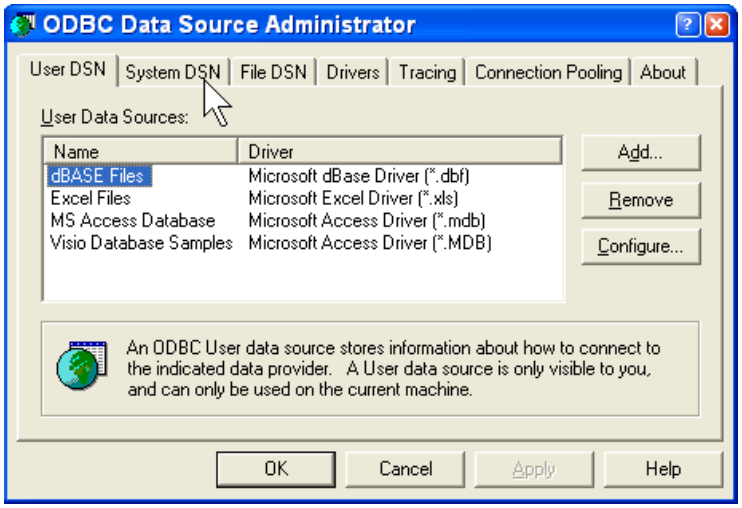

Figure 5-1: ODBC Data Source Administrator Window.

2. Select either the System DSN tab or the User DSN tab and select either Configure or Add:

   ▲ Selecting Add displays the Create New Data Source Window, as shown in Figure 5-2. Select NetezzaSQL as the driver and click Finish. The ODBC Driver Setup Window is displayed.

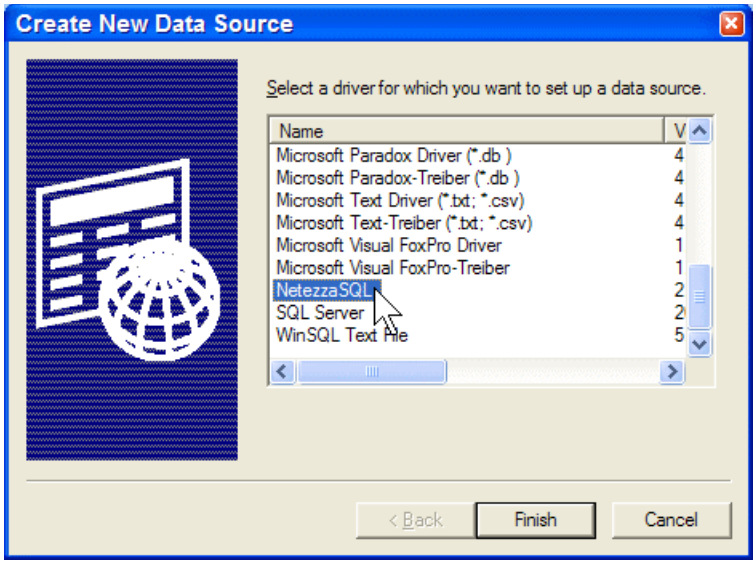

Figure 5-2: Create New Data Source Window.

   ▲ Selecting the System DSN tab displays a list of System Data Sources, as shown in Figure 5-3. Selecting Configure displays the ODBC Driver Setup Window.

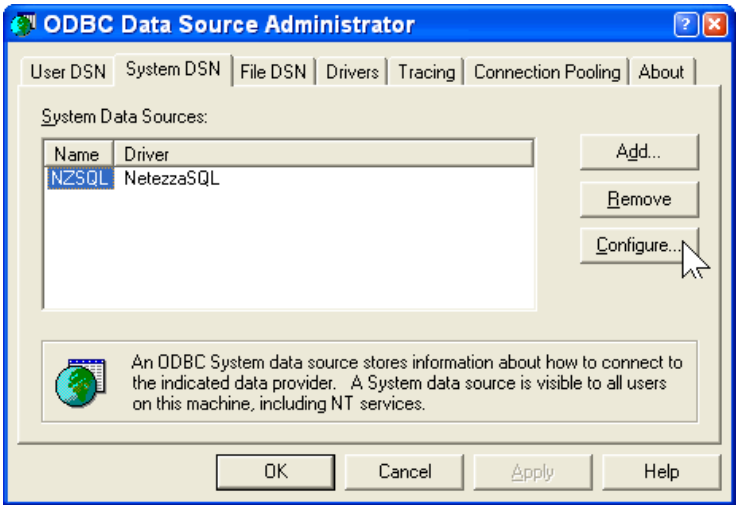

Figure 5-3: System DSN Tab.

ODBC Driver Setup Window The ODBC Driver Setup Window, as shown in Figure 5-4, Figure 5-5, Figure 5-6, and Figure 5-7 allows you to configure the DSN and Driver Options.

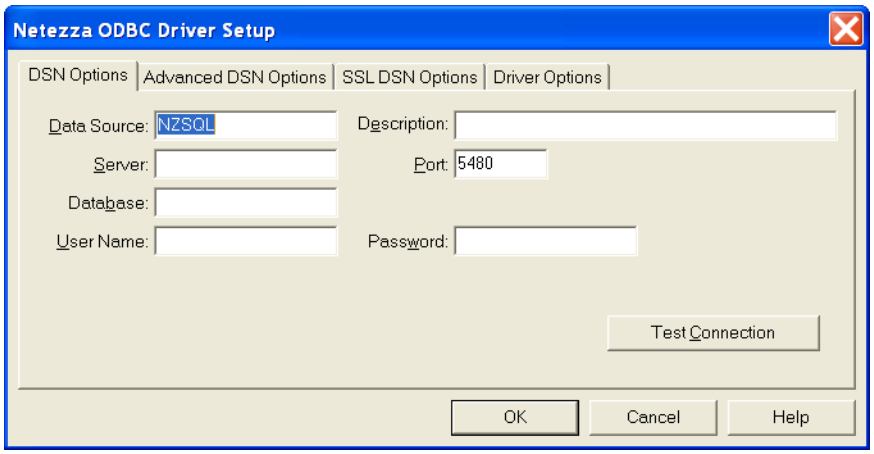

Figure 5-4: ODBC Driver Setup Window DSN Options.

Enter appropriate values for the DSN Options. If your application supplies a user name and password, these can be left blank, since your application's user name and password take precedence over the ones specified here. If the application does not supply a user name and password, supply the values. For other options, you can accept the default values, or change the options as desired.

- **Data Source** — A name that identifies the database connection properties.
- **Server** — The host name or IP address of the Netezza system to which the ODBC driver connects.
- **Database** — The name of the database on the Netezza system.
- **User Name** — The user ID of the account on the Netezza system used to access the data source.
- **Description** — A description or comment about the data in the data source.
- **Port** — The port number for the data source. The default is 5480.
- **Password** — The assigned password for the user specified.

The **Advanced Options** tab shows more options, as shown in Figure 5-5.

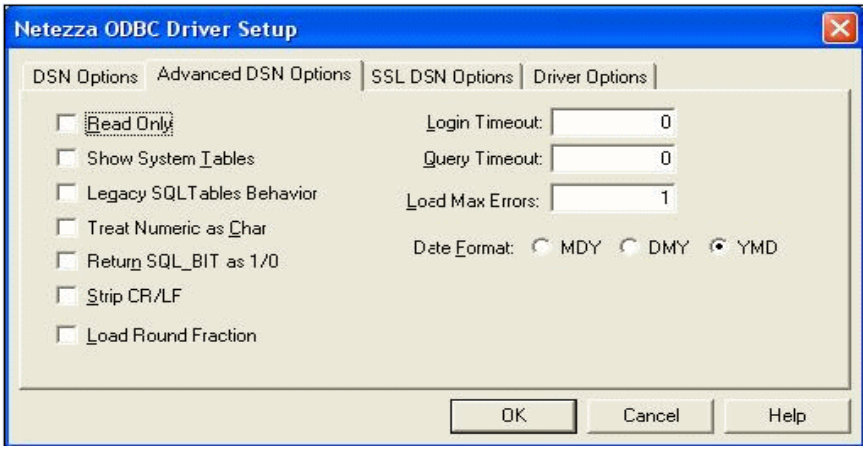

Figure 5-5: System DSN Advanced Configuration Options.

You can change these or accept the default options.

- **Read Only** — A boolean value that controls whether the database allows read-only access (only SQL Select statements are allowed), or full access. The default is deselected, which allows full access. To modify data, do not select the checkbox. If you do, only SQL Select statements are allowed.
- **Show System Tables** — When retrieving a list of tables, this boolean value controls whether the system tables are included in the list. The default is deselected to omit system tables.
- **Legacy SQL Tables Behavior** — A boolean value that, when enabled, causes SQLTables to return a list of all the users that own a database. The default is deselected to omit the list of users.
- **Treat Numeric as Char** — When selected, this boolean value causes the driver to treat SQL_C_NUMERIC buffers as SQL_C_CHAR buffers. By default, this is deselected, which means that SQL_C_NUMERIC are treated as Numeric Structures.
- **Return SQL_BIT as 1/0** — If enabled, when a return type is char or wchar, this boolean value returns the values 1 or 0 instead of t or f. The default is deselected to return t/f values. If you prefer that SQL_BIT return numeric boolean values (1/0), select the checkbox.
- **Strip CR/LF** — If enabled, this removes the carriage return and line feed characters from the SQL statements. The default is deselected to not remove these characters. Selecting the checkbox strips control characters out of your data.
- **Load Round Fraction** — If the value is true, it rounds the actual data for numeric columns whose precision exceeds the precision defined in the target table. The default is deselected, to not round.
- **Login Timeout** — The number of seconds after which to terminate the login process. Valid values are 0 (no timeout) or any positive number.
- **Query Timeout** — The number of seconds after which to terminate the query request. Valid values are 0 (no timeout) or any positive number.

- **Load Max Error**s — Tells the system to abort after *N* bad input rows. Default is 1, and 0 means no limit, with a maximum of 2,147,483,647 (the largest signed 32-bit integer).
- **Date Format** — Specifies the desired date format as one of the following values:
  - 1 — YMD. This is the default.
  - 2 — MDY
  - 3 — DMY

  Note that on Windows platforms, you specify the text value (such as YMD); on non-Windows platforms, specify the numeric value (such as 1).

The **SSL DSN Options** tab shows Secure Socket Layer options, as shown in Figure 5-6.

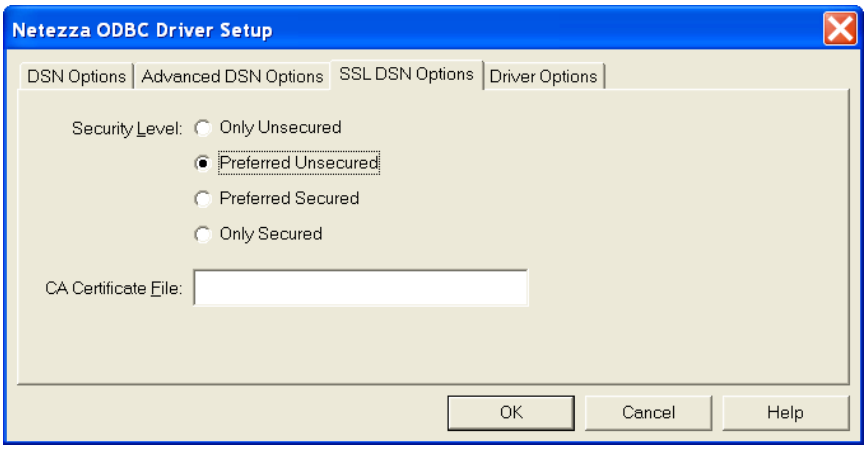

Figure 5-6: SSL DSN Options.

Select one of the following options for the Security Level, and enter a CA filename.

- **Security Level** — The level of security for the connection.
  - **Only Unsecured** — Only connections that do not use SSL to encrypt data will be accepted. This provides increased performance but carries the risks that come with an unencrypted connection.
  - **Preferred Unsecured** — Connections that do not use SSL are preferred, but a connection will not be refused solely because it uses SSL encryption.
  - **Preferred Secured** — (Default) Connections that use SSL are preferred, but a connection will not be refused solely because it does not use SSL encryption.
  - **Only Secured** — Only connections that use SSL to encrypt data will be accepted. This provides increased security but reduces performance somewhat, due to the necessity of encrypting the data stream.
- **CA Certificate File** — Enter the name of the CA (Certificate Authority) file (including the path) that will be used to authenticate connections. If a CA is being used for authentication, the name of the CA file must match the name defined by the

system administrator for the database to which you are attempting to connect. Refer to the *Netezza Administrator's Guide* for information on security and authentication.

The **Driver Options** tab allows you to configure global driver options, as shown in Figure 5-7.

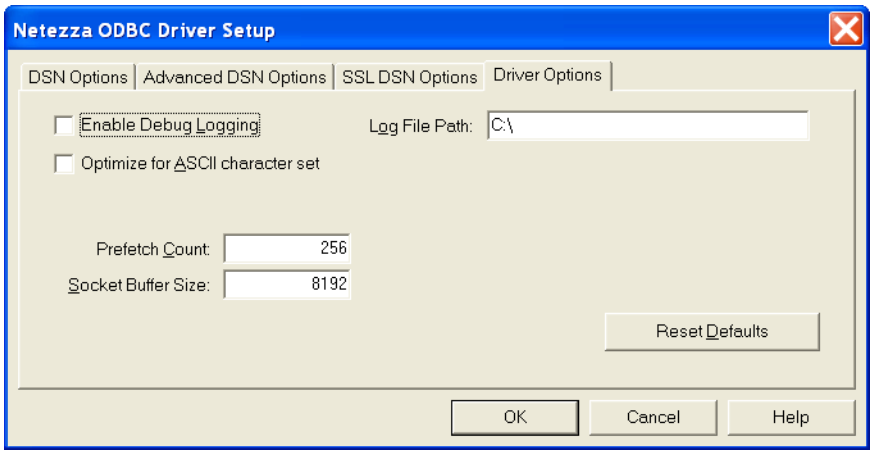

Figure 5-7: Driver Options.

- **Enable Debug Logging** — Select this box if you want the system to log information about each ODBC connection. This information may be helpful in debugging connection problems. For more information, see Appendix B, "Troubleshooting."
- **Log File Path** — The location of the debug log files. The default is c:\ on Windows systems.
- **Optimize for ASCII character set** — The Netezza appliance uses the Latin9 character encoding for char and varchar types. The character encoding for many Windows systems is similar, but not identical. If your database includes characters that use only the basic subset of letters (a-z or A-Z), numbers (0-9), or punctuation characters, select the Optimize for ASCII character set option for the driver on Windows, which will improve query performance. If your data includes special characters such as the Euro symbol or others, de-select the option to ensure that the characters convert correctly.
- **Prefetch Count** — A numeric value that sets the number of rows the driver will fetch at a time from a Netezza database. The default is 256 rows. To tune your application, set a value that optimizes network use versus memory use. Realize that the higher this value, the more memory will be required to hold these rows.
- **Socket Buffer Size** — A numeric value that specifies the size of the communications buffer in bytes. The range is 1K to 32K. The default is 8192.

Test the Connection When finished selecting DSN and Driver options, select the **DSN Options** tab and click the Test Connection button to establish a connection to this data source on your Netezza appliance server. A message box displays, indicating that the connection attempt was either successful or failed. If the connection attempt fails, check for erroneous configuration information, correct it, and test the connection again. If you cannot establish a connection after checking for errors, contact your Netezza system administrator.

Finish the Configuration A successful connection message indicates that you have successfully configured an entry for the system data source. To end the configuration process, close all the open dialog boxes.

Configuring ODBC for UNIX/Linux

This section describes the process for configuring ODBC and your Netezza appliance data source for UNIX/Linux.

You can access the files you need for the Netezza ODBC drivers and the ODBC driver installation from the Netezza FTP support site, or from the *Netezza Client* CD. The installation program (see Chapter 3, "Installing and Uninstalling Client Tools") installs the Netezza tools and ODBC libraries on your system, and creates a NetezzaSQL system data source entry (NZSQL) with appropriate default values.

After the ODBC driver is installed, the next step is to configure the environment variables to enable the ODBC Driver to locate the proper configuration files. Then you will configure the data sources.

ODBC Driver Configuration for UNIX/Linux

The driver manager and the Netezza ODBC drivers use a configuration file named **.odbcinst.ini** to get information about ODBC drivers. During the Netezza ODBC driver configuration process, you supply configuration information for the Netezza ODBC driver by modifying entries in the **.odbcinst.ini** file.

Note: The period precedes the file name only when the file is located in the user's home directory as a hidden file. For a description of hidden and non-hidden files and how they are used, refer to Appendix A, "ODBC Configuration Files."

The following shows the contents of a 32-bit file named **odbcinst.ini.sample**. This file is in the **nz/lib** directory (ordinarily **/usr/local/nz/lib**), after you have installed the Netezza driver. Use the contents of the sample file to configure your own **.odbcinst.ini** file.

```
[ODBC Drivers]
NetezzaSQL = Installed
[ODBC Connection Pooling]
PerfMon          = 0
Retry Wait       =

[NetezzaSQL]
Driver           = /usr/local/nz/lib/libnzodbc.so
Setup            = /usr/local/nz/lib/libnzodbc.so
APILevel         = 1
ConnectFunctions = YYN
Description      = Netezza ODBC driver
DriverODBCVer    = 03.51
DebugLogging    = false
LogPath         = /tmp
UnicodeTranslationOption = utf8
CharacterTranslationOption = all
PreFetch          = 256
Socket            = 8192

[ODBC]
```

```
Trace            = 0
TraceAutoStop    = 0
TraceFile        = /tmp/sql.log
```

Note: The Netezza ODBC driver properties are explained in Table 4-1 on page 4-2. Use this table to select appropriate values for the configuration entries in your .odbcinst.ini file. The CharacterTranslationOption can be set to “all’ or “latin9.”

ODBC Driver Configuration Steps for UNIX/Linux

The following are the steps to configure a Netezza ODBC driver on a UNIX/Linux client system:

1. Log in, using your own user ID and password.
2. Modify environment variables.
3. Create ODBC Driver Configuration file.
4. Optionally modify ODBC Driver Configuration file.

ODBC Driver Configuration Procedure for UNIX/Linux

The following sections describe each of the steps in the ODBC driver configuration procedure.

1. If you previously installed the Netezza client software while logged in as the superuser, you can now log out and log back in using your own user ID and password. You can use your own user ID and password when you configure your ODBC driver and Netezza appliance data sources. Configuring the ODBC driver and Netezza appliance data sources while logged in as an ordinary user allows each user of the system to supply different configuration information to access different Netezza appliance data sources.
2. Modify the environment variables to include the appropriate installation directories. The following are command examples:

   For AIX:

   ```
   export PATH=$PATH:/usr/local/nz/bin
   export LIBPATH=$LIBPATH:/usr/local/nz/lib
   ```

   For HP-UX:

   ```
   export PATH=$PATH:/usr/local/nz/bin
   export SHLIB_PATH=$SHLIB_PATH:/opt/datanz/nz/bin
   ```

   For Linux, Solaris:

   ```
   export PATH=$PATH:/usr/local/nz/bin
   export LD_LIBRARY_PATH=$LD_LIBRARY_PATH:/usr/local/nz/lib
   ```

   Note: The above commands change the environment for the current session only. After you have verified that your environment modifications work correctly, edit your login script to set the environment variables when users log in.
3. Configure Netezza ODBC drivers by using a text editor to modify the contents of the **`.odbcinst.ini`** configuration file. You normally place the **`.odbcinst.ini`** file in your home directory while logged in as an ordinary user.

You may want to place the **.odbcinst.ini** file in a different location, and you may want to do the configuration as the superuser, rather than as an ordinary user. These issues are discussed in Appendix A, "ODBC Configuration Files."

This step assumes you are logged in as an ordinary user and are placing the **.odbcinst.ini** file in your home directory. What you do in this step depends on whether or not the **.odbcinst.ini** file already exists in your home directory.

- **The Configuration File Exists.** Use a text editor, such as **vi**, to examine the **.odbcinst.ini** configuration file if it already exists in your home directory, which may occur for different reasons, including the following:
  - Your client system was previously configured for one or more ODBC drivers other than the Netezza ODBC driver. If you still want to use those ODBC drivers, do not modify existing entries in the **.odbcinst.ini** file. In this case, add the entry in the **odbcinst.ini.sample** file to the end of the existing contents of the **.odbcinst.ini** file.
  - The Netezza client software and a Netezza ODBC driver were previously installed on your client system, and the Netezza ODBC driver has already been configured. If this is so, you may not have to add any entries to the **.odbcinst.ini** file. If your **.odbcinst.ini** file already contains an entry similar to that found in the **odbcinst.ini.sample** file, do nothing. If it does not, then add the entry in the **odbcinst.ini.sample** file to the end of the existing contents of the **.odbcinst.ini** file.
- **The Configuration File Does Not Exist.** If the **.odbinst.ini** configuration file does not yet exist in your home directory, copy the entire **odbcinst.ini.sample** file into your home directory and rename it **.odbcinst.ini**.

4. The entry from the **odbcinst.ini.sample** file can usually be used without modification in your **.odbcinst.ini** file. Consult your Netezza system administrator to see if any configuration entries need to be changed for your installation.

ODBC Data Source Configuration for UNIX/Linux

The driver manager and the Netezza ODBC drivers use the **.odbc.ini** configuration file to get information about data sources. To configure data sources on a UNIX/Linux system, use a text editor to modify the **.odbc.ini** file.

Data Source Configuration File

This section describes the **.odbc.ini** file and shows samples of its contents. You normally place the **.odbc.ini** file in your own home directory. It specifies details about the Netezza appliance data sources you want to access.

Note: The period precedes the file name only when the file is located in the user's home directory.

To configure the **.odbc.ini** file entries to access the system data source on your own Netezza appliance, get the following configuration information from your Netezza system administrator, and substitute it for the information in **odbc.ini**:

- **Database** — The name of the system database on the Netezza appliance.
- **Username** — A valid user ID for accessing the system data source.

- **Servername** — The server host name or IP address of your Netezza appliance.
- **Password** — A valid password for accessing the system data source.

The following shows the contents of a file named **odbc.ini.sample**, which is in the **nz/lib** directory after you have installed your ODBC drivers. The sample file describes a system data source found on every Netezza appliance, and includes comments explaining each parameter. Use the contents of this sample file, along with the administrator information you substituted, to configure your own **.odbc.ini** file.

```
;
;  odbc.ini
;
[ODBC Data Sources]
NZSQL = NetezzaSQL


[NZSQL]
;Path of a driver used for database connection
Driver                = /usr/local/nz/lib/libnzodbc.so

;A decription used to identify the database connection properties.
Description           = NetezzaSQL ODBC

;The name of a machine or IP address where the database is located.
Servername            = 127.0.0.1

;The port number to connect.
Port                  = 5480

;The name of a database.
Database              = dev

;The name of a valid user.
Username              = guest

;The password for  the user.
Password              = password

;Only Select SQL statements will be allowed if this field is checked.
ReadOnly              = false

;When set to true, system tables will be included in the available
table list.
ShowSystemTables      = false
```

```
;When set to true, use old (2.5) functionality for returning schemas in
SQLTables.
LegacySQLTables       = false

;Specifies login timeout in seconds.
LoginTimeout          = 0

;Specifies query timeout in seconds.
QueryTimeout          = 0

;Specifies date format as follows -
; 1: YMD
; 2: MDY
; 3: DMY
DateFormat            = 1

;When selected, driver treats SQL_C_NUMERIC buffers as SQL_C_CHAR
buffers.
NumericAsChar         = false

;Return SQL_BIT as "1"/"0" when char.
SQLBitOneZero         = false

;When enabled, replaces all occurences of CRLF, CR and LF with a single
space.
StripCRLF             = false

;Specifies level of security on the system
securityLevel         = preferredUnSecured

;Name of the Root CA certificate.
caCertFile            =

;Specifies maximum errors allowed. 0 value means no limit on
loadMaxErrors value.
;loadMaxErrors         = 1

;If value is true, it rounds the data of numeric columns whose
precision exceeds the precision defined.
;loadRoundFraction     = 0
```

```
[ODBC]
IANAAppCodePage=4
InstallDir=/opt/odbc32v51
Trace=0
TraceDll=/opt/odbc32v51/lib/odbctrac.so
TraceFile=odbctrace.out
UseCursorLib=0
```

Note: The data source properties are explained starting on page 5-6. Use this information to select appropriate values for the entries in your **.odbc.ini** file.

Data Source Configuration Steps for UNIX/Linux

The following are the steps to configure Netezza appliance data sources on a UNIX/Linux client system:

1. Create the Data Source Configuration File.
2. Configure the Data Source.

Data Source Configuration Procedure for UNIX/Linux

The following sections describe each of the steps to configure the data source.

1. Configure Netezza appliance data sources by modifying the **.odbc.ini** configuration file. As with the **.obdcinst.ini** file, you normally place the **.odbc.ini** configuration file in your home directory while logged in as an ordinary user.

   Note: In some cases, you may want to place the **.odbc.ini** file in another location, and you may want to do the configuration as the superuser rather than as an ordinary user. These issues are discussed in Appendix A, "ODBC Configuration Files."

   This step assumes you are logged in as an ordinary user and are placing the **.odbc.ini** file in your home directory. What you do in this step depends on whether or not the **.odbc.ini** file already exists in your home directory.

   - **The Configuration File Exists.** If the **.odbc.ini** configuration file already exists in your home directory, use a text editor to examine the contents of the **.odbc.ini** file to see if it already contains entries for the Netezza appliance data sources to access. If it does not, copy the contents of the **odbc.ini.sample** file to the end of your existing **.odbc.ini** file. Do not modify any existing entries in the file.
   - **The Configuration File Does Not Exist.** If the **.odbc.ini** configuration file does not yet exist in your home directory, copy the entire **odbc.ini.sample** file into your home directory and rename it **.odbc.ini**.

2. Once you have initially set up the **.odbc.ini** configuration file, use a text editor to modify entries in the file to describe the Netezza appliance data sources to access, as described in the section "Data Source Configuration File" on page 5-12. To access additional Netezza appliance data sources, supply a set of configuration entries, similar to that found in the **odbc.ini.sample** file, for each data source to access.

ODBC Connectivity Validation for UNIX/Linux

To validate that you can use ODBC to connect to an Netezza appliance data source, use the interactive SQL query program called **isql**. The **isql** program is installed on your system when you install the unixODBC driver manager (See "unixODBC Driver Manager Installation" on page 4-4.) You can use **isql** to run SQL queries against both the system data source and any user data sources you configure.

Note: If you are using a driver manager other than the unixODBC driver manager, consult your Netezza system administrator to see if there is an SQL query program installed on your client system that you can use instead of **isql** to validate ODBC connectivity and query Netezza appliance data sources.

The following example uses the DataDirect driver manager, where you run the **example** program in the **example** directory, specifying the NZSQL DSN, login, and password, displaying output as in the following:

```
./example
   DataDirect Technologies, Inc. ODBC Example Application.
   Enter the data source name : NZSQL
   Enter the user name        : admin
   Enter the password         : password
   Enter SQL statements (Press ENTER to QUIT)
SQL>
```

Once connected, you can run the same queries as used in the following examples. The system should return the same results.

ODBC Connectivity Validation Steps for UNIX/Linux

The following are the steps to validate ODBC connectivity to the Netezza appliance data sources configured in the previous section.

1. Change directories.
2. Connect to NZSQL system data source.
3. Query system database.
4. Display query results.
5. Connect to Test User data source.
6. Query user database.
7. Display query results.

ODBC Connectivity Validation Procedure for UNIX/Linux

The following sections describe each of the above steps in the connectivity validation procedure.

1. Change to the directory containing the **isql** interactive SQL application by entering the following command:

   ```
   cd /usr/local/unixODBC/bin
   ```

2. Start the **isql** interactive SQL application, and connect to the **NZSQL** system data source by entering the following command:

```
isql -v NZSQL
```

If your connection is successful, you will see a response similar to the following:

```
+---------------------------------------+
| Connected!                            |
|                                       |
| sql-statement                         |
| help [tablename]                      |
| quit                                  |
+---------------------------------------+
```

3. The **system** database in the **NZSQL** data source contains a table named **_t_dslice**. Enter the following command to query the **_t_dslice** table:

```
select * from _t_dslice
```

4. If your ODBC driver is correctly installed, and your system data source is correctly configured, the **isql** program displays the contents of the **_t_dslice** table from the **system** database in the **NZSQL** data source. You will see a response similar to the following:

```
+------------+------------+------------+------------+
| ds_id      | ds_pridskid| ds_priptid | ds_mirdskid|
+------------+------------+------------+------------+
| 1          | 1002       | 0          |            |
+------------+------------+------------+------------+
```

5. Enter the **quit** command to end the connection to the NZSQL system data source and exit the **isql** program. Then restart **isql** and connect to the **Test_User** user data source by entering the following command:

```
isql -v Test_User
```

If your connection is successful, you will again see a response similar to the following:

```
+---------------------------------------+
| Connected!                            |
|                                       |
| sql-statement                         |
| help [tablename]                      |
| quit                                  |
+---------------------------------------+
```

6. The example assumes the **system** database in the **Test_User** data source contains a table named **a_inventory**. Enter the following command to query the **a_inventory** table:

```
select * from a_inventory
```

7. If the ODBC driver is correctly installed, and the user data source is correctly configured, the **isql** program displays the contents of the **a_inventory** table from the **system** database in the **Test_User** data source:

```
+------------+----------+------------+
| part_no    | part_name| on_hand    |
+------------+----------+------------+
| 181        | Wheel    | 449        |
| 124        | Bolt     | 900        |
| 105        | Gear     | 0          |
| 134        | Nut      | 743        |
+------------+----------+------------+
```

CHAPTER 6

Installing and Configuring JDBC

What's in this chapter

- Installing JDBC for Windows
- Installing JDBC for UNIX/Linux
- Configuring a JDBC Data Source (Windows Only)

This chapter describes the procedures to install and configure JDBC.

You can have only one JDBC driver installed on your system at any given time. If you need a different driver, you will need to remove the installed driver and install the different one. For more information on removing the driver, see Chapter 3, "Installing and Uninstalling Client Tools."

Note: After installation, when you run an application trying to use JDBC, you are prompted for a value called Class Name. For the Netezza Driver, enter the following:

```
org.netezza.Driver
```

Installing JDBC for Windows

This section describes the procedure for installing Netezza JDBC components on a Windows system.

JDBC Driver Installation Steps for Windows

The following are the steps for installing the JDBC driver with the interactive GUI:

1. Launch JDBC Driver Installation Program.
2. Select the language for the installation program.
3. Review the Introduction window.
4. Choose the Install Folder.
5. Review the installation options.
6. Finish the installation.

JDBC Driver Installation Procedure for Windows via the GUI

This section describes each of the steps to install the Netezza JDBC driver on your system. The examples are for a Windows XP system for version 5.0.

Complete the following steps to install the JDBC driver.

1. Start the installation program by double-clicking the **nzjdbcsetup.exe** program. For directions, see "Installation Files Access" on page 1-5.
2. A language option window is displayed, which controls the language for the installation program. Accept the default of English and click **OK**.
3. The Introduction window is displayed. Review the information and click **Next**.
4. The Choose Install Folder window is displayed. This allows you the option to change the installation folder from the Netezza default. When done, click **Next**.
5. The Pre-Installation Summary window is displayed, summarizing your installation choices. To page through the previous selections to review selections, use the **Previous** button. When finished, click **Install**.
6. The Installation Completed window indicates that the Netezza JDBC driver installation process has completed successfully. Click **Done** to exit.

Installing JDBC for UNIX/Linux

Before you can configure a Netezza JDBC driver on your system, you must first install the Netezza client tools software, which includes the JDBC driver. If you have not yet done that, do the procedure described in Chapter 3, "Installing and Uninstalling Client Tools."

JDBC Driver Installation for UNIX/Linux on a Supported Platform

The Netezza appliance client tools installation procedure copies the Java **jar** file containing the JDBC driver into the nz/lib directory (usually **/usr/local/nz/lib**). This file is named **nzjdbc3.jar**. To configure JDBC on a Linux client system, run an **export** command to include an appropriate entry in your **CLASSPATH** environment variable.

Note: Before you use JDBC, verify that your system has Java installed. For more information, go to **http://www.java.com** where you can test your system for Java software support and download the latest version for your system.

Run the following command to modify your **CLASSPATH** environment variable to include the location of the Netezza JDBC Version 3.0 driver:

```
export CLASSPATH=.:/usr/local/nz/lib/nzjdbc3.jar
```

Note: The above **export** commands change the environment for the current session only. Once you have verified that your environment modification works correctly, you can edit your login script to set the environment variable when users log in.

JDBC Driver Installation for UNIX/Linux on an Unsupported Platform

This section describes how to install a Netezza JDBC driver on a Linux/UNIX systems that is not one of the supported Linux/UNIX platforms as documented in Chapter 2, "Support and Release Compatibility."

JDBC Driver Installation Steps for UNIX/Linux on an Unsupported Platform

The following are the steps to install a JDBC driver on an unsupported Linux/UNIX system:

1. Change Directories on Netezza Appliance Host.

2. Copy Driver to Client System.
3. Modify Environment Variable.

JDBC Driver Installation Procedure for UNIX/Linux on an Unsupported Platform

The following sections describe each of the above steps in the procedure for installing the JDBC driver on the client system.

1. Change to the appropriate directory on your Netezza appliance host:

```
cd /nz/kit/sbin
```

2. Use the **rcp** or **ftp** command (using binary transfer mode) to copy the JDBC driver to your client machine. The following is a command example:

```
rcp nzjdbc3.jar <client machine>
```

3. Modify the **CLASSPATH** environment variable on the client machine to include the path of the directory in which you placed the JDBC driver.

Configuring a JDBC Data Source (Windows Only)

This section describes the procedure for configuring a JDBC data source on a Windows system. You can save a data source name (DSN) on the local server through the use of the nzjdbc.ini file, which is in the same directory as the driver itself, and which controls the JDBC Graphical User Interface (GUI).

JDBC Data Source Configuration Steps

The following are the steps for configuring the JDBC data source:

1. Start the GUI.
2. Add a data source.
3. Enter Connection Options.
4. Enter Advanced Options.
5. Enter SSL Options.
6. Test connection.
7. Verify data source.

JDBC Data Source Configuration Procedure

This section describes each of the steps in the procedure for configuring and saving the JDBC DSN in version 4.6. Your displayed version may be different.

1. Start the GUI by running the following:

```
java -jar <driver JAR>
```

2. The list of data sources screen is displayed. Select Add to add a new data source.

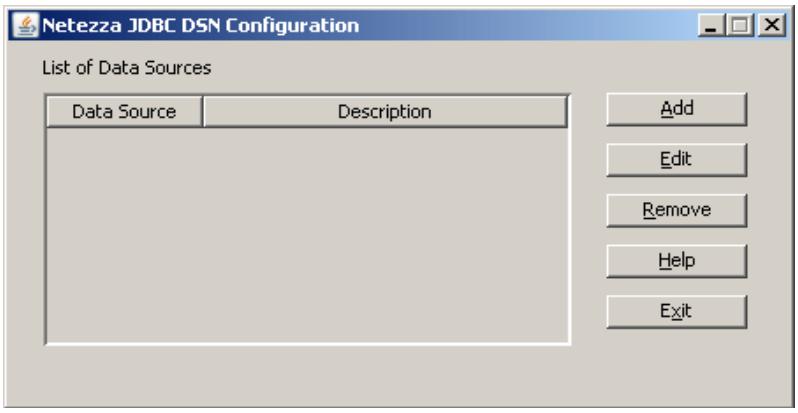

Figure 6-1: List of Data Sources screen.

3. The Datasource Configuration screen is displayed. Select the Connection Options tab and enter the information in the appropriate fields:

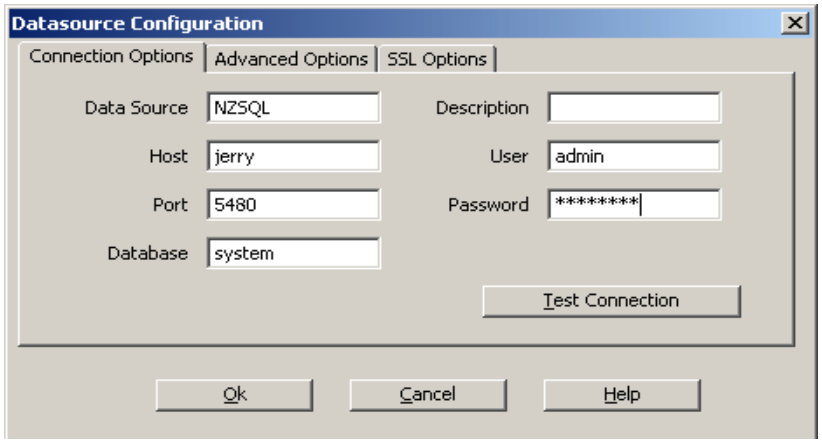

Figure 6-2: Datasource Configuration Connection Options.

- Data Source— Name of the data source (Mandatory).
- Host— Name of the host to connect to (Mandatory).
- Port— Port number (Mandatory).
- Database— Database (Catalog) name (Mandatory).
- Description— Description for the data source.
- User— Username to connect to the database.
- Password— Password for authentication.
- Test Connection— Tests the connection with the given parameters.

4. Select the Advanced Options tab and enter the information in the appropriate fields:

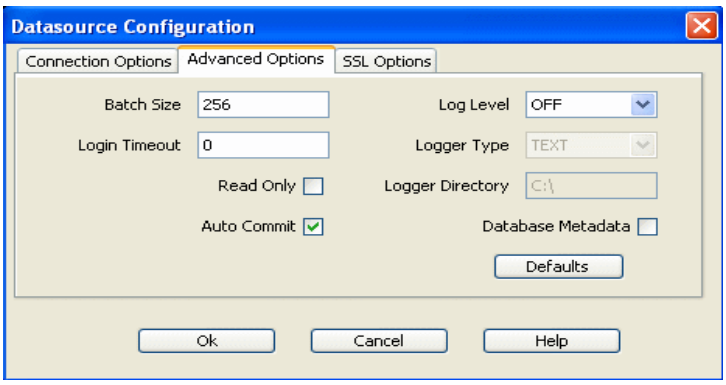

Figure 6-3: Datasource Configuration Advanced Options.

- Batch Size— Number of rows to be fetched in a cycle. Default - 256.
- Login Timeout— Maximum wait time to make a connection. Default - infinite.
- Read Only— Read Only property for the connection. Default - false.
- Auto Commit— Auto Commit property for the connection. Default - true.
- Log Level— Log Level for the connection. Default - OFF.
- Logger Type— The type to be used for logging. Default - TEXT.
- Logger Directory— Log directory where the log files will be dumped.
- Database Metadata— When selected, you can get metadata information that is not restricted to the current catalog/schema. Not selecting this means your information will be restricted to the current catalog/schema. Default - not selected.
- Defaults— Selects the defaults for these options.

5. Select the SSL Options tab and enter the security level information as appropriate:

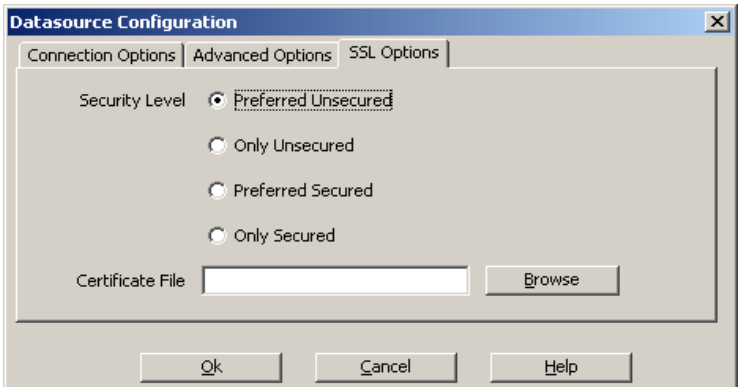

Figure 6-4: Datasource Configuration SSL Options.

- **Preferred Unsecured** — Connections that do not use SSL are preferred, but a connection will not be refused solely because it uses SSL encryption.
- **Only Unsecured** — Only connections that do not use SSL to encrypt data will be accepted. This provides increased performance but carries the risks that come with an unencrypted connection.
- **Preferred Secured** — Connections that use SSL are preferred, but a connection will not be refused solely because it does not use SSL encryption.
- **Only Secured** — Only connections that use SSL to encrypt data will be accepted. This provides increased security but reduces performance somewhat, due to the necessity of encrypting the data stream.

CA Certificate File — Enter the name of the CA (Certificate Authority) file used to authenticate connections. If a CA is being used for authentication, the name of the CA file must match the name defined by the system administrator for the database to which you are attempting to connect. Refer to the *Netezza Administrator's Guide* for information on security and authentication.

6. When finished entering Datasource Configuration options, test the connection by selecting the Test Connection button on the Connection Options tab. When the connection is established, you will see a confirmation screen. Select OK.
7. When all your options have been entered and your connection has ben tested, select Ok on the Datasource Configuration screen. The Data Source and description will appear in the List of Data Sources screen.

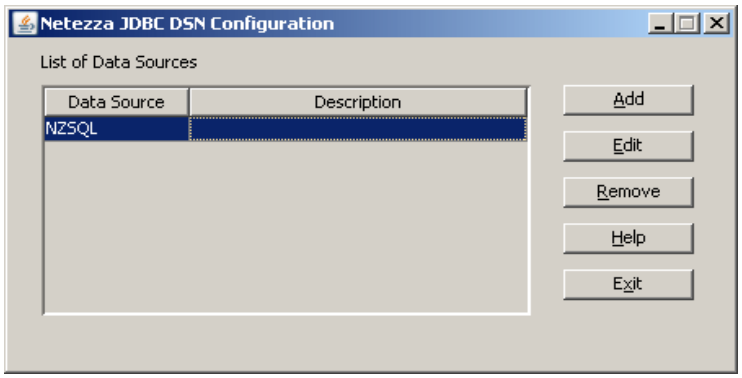

Figure 6-5: List of Data Sources.

8. Select Exit when finished.

CHAPTER 7
Installing and Configuring OLE DB

What's in this chapter

- Installing OLE DB Provider
- Configuring an OLE DB Data Source

This chapter describes the procedure for installing the Netezza OLE DB provider on a Windows client system, configuring the provider and data sources, and testing for connectivity to Netezza appliance data sources.

Installing OLE DB Provider

This section describes the steps and procedure for installing the Netezza OLE DB provider on a Windows client system.

The OLE DB provider offers "silent" installation options for both 32-bit and 64-bit installation executables. The installation options follow:

- For 32-bit installations:
  - **c:\>nzoledbsetup.exe** (full GUI install)
  - **c:\>nzoledbsetup.exe -i silent** (silent install, VC8 Libraries not installed)
  - **c:\>nzoledbsetup.exe -i silent -DVC8_LIB_INSTALL=1** (silent install, VC8 Libraries installed)
- For 64-bit installations:
  - **c:\>nzoledbsetup64.exe** (full GUI install)
  - **c:\>nzoledbsetup64.exe -i silent** (silent install, VC8 Libraries not installed)
  - **c:\>nzoledbsetup64.exe -i silent -DVC8_LIBS_INSTALL=1** (silent install, VC8 Libraries installed)

OLE DB Provider Installation Steps

The following are the steps for installing a Netezza OLE DB provider on a Windows client system:

1. Launch the OLE DB provider installation program.
2. Select the language for the installation program.
3. Review the Introduction window.

4. Install the Visual Studio 8 libraries.
5. Review the installation options.
6. Finish the installation.

OLE DB Provider Installation Procedure

This section describes each of the steps in the procedure for installing a Netezza OLE DB provider. The examples are for a Windows XP system for version 5.0. The procedure is the same on other Windows systems.

Complete the following steps to install the OLE DB provider.

1. Start the installation program by double-clicking the **`nzoledbsetup.exe`** program. For directions, see "Installation Files Access" on page 1-5.
2. A language options window is displayed, which controls the language for the installation program. Accept the default of English and click **OK**.
3. The Introduction window is displayed. Review the information and click **Next**.
4. The OLE DB provider requires the Visual Studio 8 (VC8) Libraries on the client system. If the installation program finds the VC8 libraries, it uses the installed files and proceeds to the Pre-Installation Summary window.

   However, if it cannot find the VC8 files on the system, it displays the Visual Studios 8 Libraries window. Select **Install** and click **Next**. If you are sure that the VC8 libraries are present on the system even though the installer could not locate them, select **Do not install** and click **Next**.

   If the driver does not function properly because the libraries are missing or not found, run the driver installation program again to install the VC8 libraries.
5. The Pre-Installation Summary window is displayed, summarizing your installation choices. To change any of your choices, use the **Previous** button to page through the previous windows. When finished, click **Install** to complete the installation process.

   After you click **Install**, the installation program does the following:

   - Installs the Netezza OLE DB provider. A 32-bit provider on a 32-bit system and a 64-bit provider on a 64-bit system are installed in the **`\WINDOWS\system32`** folder. On a 64-bit system, both providers are installed, and the 32-bit provider is installed in the **`\WINDOWS\SysWOW64`** folder.
   - Places entries in the Windows registry to add the Netezza OLE DB Provider to the provider list.
6. The Install Complete window is displayed, which indicates that the installation process has completed successfully. Click **Done** to close the installation program.

   Before trying to access an OLE DB data source, you must configure it. The next section describes the configuration procedure.

Configuring an OLE DB Data Source

This example shows you how to configure connectivity to the database named *system* on the server named *npsmachine*.

OLE DB Data Source Configuration Validation Steps

The following are the steps to validate connectivity to the database:

1. Create an empty .udl file.
2. Open the **Data Link Properties** dialog box.
3. Select **Netezza OLE DB Data Provider**.
4. Specify the Netezza system.
5. Enter User ID and Password.
6. Select a database catalog to connect to.
7. Test the database connection.
8. Verify the connection.
9. In case of an unsuccessful connection, review parameters.

The following sections describe each of the steps in the above procedure to validate ODBC connectivity to the system data source using WinSQL.

OLE DB Data Source Configuration Validation Procedure

The following is the procedure to validate connectivity to the database:

1. Using a text editor, create an empty text file and save it to the desktop as `test.udl` .
2. Double-click on the desktop icon for the `test.udl` file that you just created. A **Data Link Properties** dialog box appears, as shown in Figure 7-1.

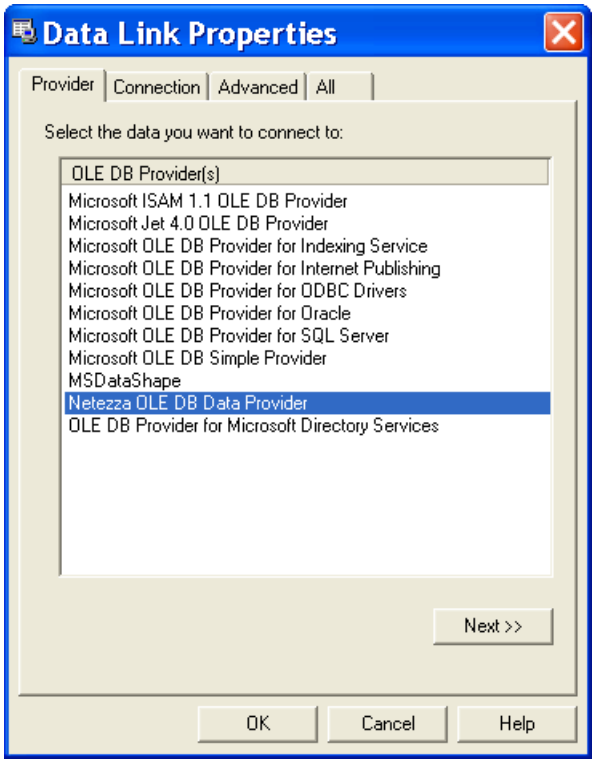
Data Link Properties
Provider
Connection
Advanced
All
Select the data you want to connect to:
OLE DB Provider(s)
Microsoft ISAM 1.1 OLE DB Provider
Microsoft Jet 4.0 OLE DB Provider
Microsoft OLE DB Provider for Indexing Service
Microsoft OLE DB Provider for Internet Publishing
Microsoft OLE DB Provider for ODBC Drivers
Microsoft OLE DB Provider for Oracle
Microsoft OLE DB Provider for SQL Server
Microsoft OLE DB Simple Provider
MSDataShape
Netezza OLE DB Data Provider
OLE DB Provider for Microsoft Directory Services
Next >>
OK
Cancel
Help

Figure 7-1: OLE DB Provider: Selection Tab.

3. In the list of OLE DB Providers, select **Netezza OLE DB Data Provider** (which was installed as part of the OLE DB Provider installation procedure) and click **Next**. The OLE DB Connection tab appears, as in Figure 7-2.

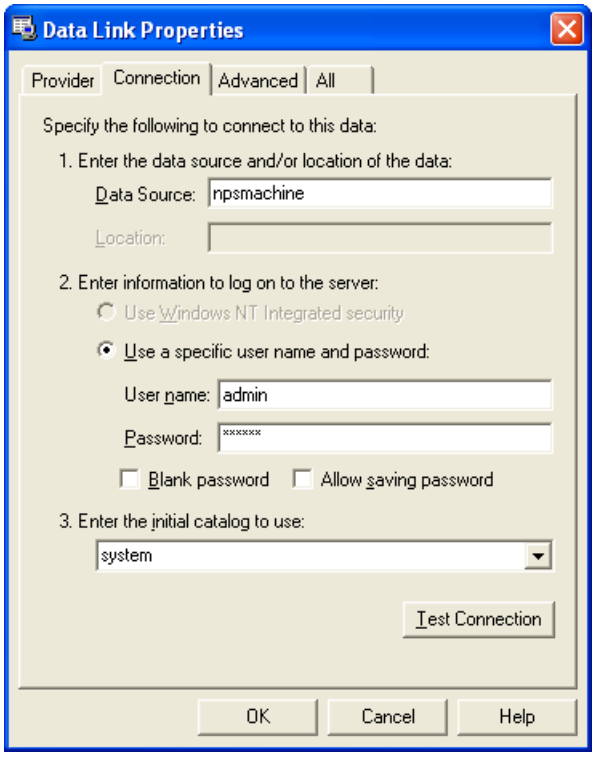

Figure 7-2: OLE DB Provider: Connection Tab.

4. In the Data Source box, specify the name of the Netezza system on which the database resides.
5. Specify appropriate values for User ID and Password.
6. From the drop-down list, choose the name of the database catalog to connect to.
7. Click the **Test Connection** button in the lower right corner of the Connection Tab.
8. The system authenticates the user, establishes the connection, and, if the connection is successful, displays the dialog shown in Figure 7-3.

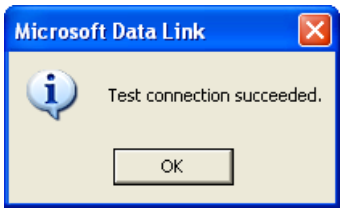

Figure 7-3: OLE DB Provider: Connection Succeeded Message.

9. If the connection fails, verify that the SSL security parameters agree with those defined by the system administrator for the system to which you are attempting to connect. These parameters are shown as an example in Figure 7-4.

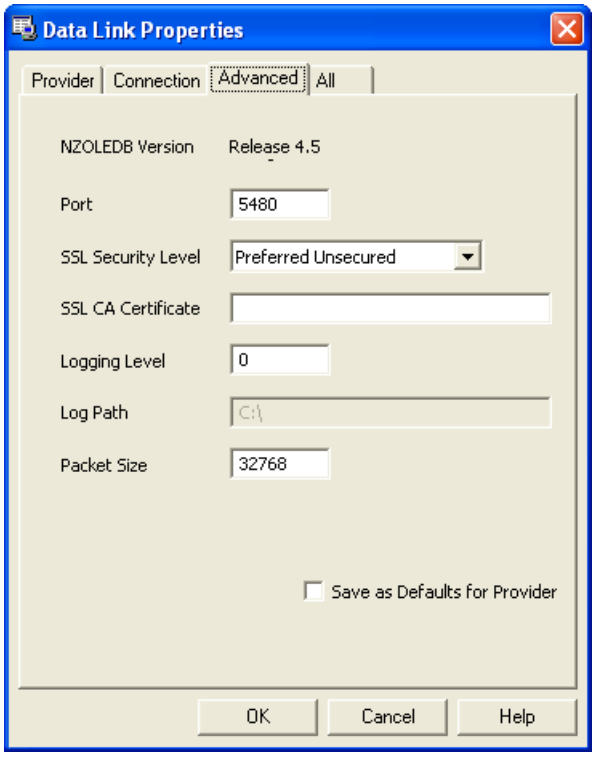

Figure 7-4: OLE DB Provider: Advanced Tab.

APPENDIX A

ODBC Configuration Files

What's in this appendix

- Netezza ODBC Driver
- unixODBC Driver Manager
- DataDirect Driver Manager
- Investigating Client System Configuration

This appendix introduces advanced considerations for configuring ODBC drivers and data sources on Linux/UNIX client systems, with respect to locating the **`.odbcinst.ini`** and **`.odbc.ini`** configuration files.

Netezza ODBC Driver

This section discusses the names and locations of the two ODBC configuration files. You can configure the two ODBC configuration files as *hidden* files or as *non-hidden* files, and you can place them in directories other than in the user's home directory.

The ODBC configuration examples for Linux/UNIX systems in Chapter 5 configure the two ODBC configuration files as hidden files by preceding their names with an initial "**.**" character. You can alternatively configure them as non-hidden files by omitting the initial "**.**" character.

- **Hidden Files.** When you place the two ODBC configuration files in the user's home directory, the Netezza ODBC driver expects the files to have the hidden file names **`.odbcinst.ini`** and **`.odbc.ini`**.
- **Non-Hidden Files.** When you place the two ODBC configuration files in a directory other than the user's home directory, the Netezza ODBC driver expects the files to have the non-hidden file names **`odbcinst.ini`** and **`odbc.ini`**.

ODBC Driver Search Behavior

The default behavior of the Netezza ODBC driver is to search the following places, in sequence, to locate the two ODBC configuration files:

1. If there is an **`NZ_ODBC_INI_PATH`** entry in the environment describing a directory path, the driver looks first in that directory for two non-hidden configuration files.
2. The driver then looks in the current user's home directory for the two hidden files: **`.odbcinst.ini`** and **`.odbc.ini`**. If the two configuration files exist as non-hidden files in the user's home directory, the driver ignores them.

3. If the driver does not find hidden configuration files in the current user's home directory, it next looks in the **/usr/local/etc** directory for the two non-hidden files **odbcinst.ini** and **odbc.ini**. If the two configuration files exist as hidden files in the **/usr/local/etc** directory, the driver ignores them.

ODBC Driver Considerations

Placing hidden configuration files in the user's home directory is useful if each user of the Linux/UNIX system has a separate set of configuration files, possibly describing different Netezza appliance data sources. Each user's home directory has a pair of hidden configuration files describing the Netezza appliance data sources that user can access.

Placing non-hidden configuration files in either the first or third location is useful if you want all users of the client system to have access to the same set of Netezza appliance data sources, without requiring a separate set of configuration files in each user's home directory. The
/usr/local/etc directory or the **NZ_ODBC_INI_PATH** directory contains a single set of non-hidden configuration files describing the Netezza appliance data sources that all users of the client system can access.

You can decide where to place the ODBC configuration files depending on how you are using the client system.

The way in which the Netezza ODBC driver looks for ODBC configuration files is straightforward and unambiguous. However, the driver manager installed on the client system also uses the two ODBC configuration files. The way in which a Netezza ODBC driver looks for ODBC configuration files can sometime conflict with the way in which the driver manager looks for them. This can sometimes cause configuration problems.

unixODBC Driver Manager

This section describes how the unixODBC driver manager searches for the ODBC configuration files and how the unixODBC driver manager interacts with the Netezza ODBC drivers.

unixODBC Driver Manager Search Behavior

The default behavior of the unixODBC driver manager is to search the following places, in sequence, to locate the two ODBC configuration files:

1. The driver manager looks first in the current user's home directory for the two *hidden* files: **.odbcinst.ini** and **.odbc.ini**. If the two configuration files exist as non-hidden files in the user's home directory, the driver manager ignores them.
2. If the driver manager does not find hidden configuration files in the user's home directory, it looks next in the **/usr/local/etc** directory for the two *non-hidden* files **odbcinst.ini** and **odbc.ini**. If the two configuration files exist as hidden files in the **/usr/local/etc** directory, the driver manager ignores them.

Compatibility

The methods the Netezza ODBC driver and the unixODBC driver manager use to locate the configuration files are compatible. It is relatively easy to ensure that the Netezza ODBC Driver and the unixODBC driver manager find the two ODBC configuration files in the same place. Their uses of hidden and non-hidden files are also compatible.

Place hidden configuration files in each user's home directory if each user needs to access different Netezza appliance data sources. Place non-hidden configuration files in the **/usr/local/etc** directory if all users will access the same Netezza appliance data sources.

With the unixODBC driver manager, you do not need to use the **NZ_ODBC_INI_PATH** environment variable.

DataDirect Driver Manager

This section describes how the DataDirect driver manager searches for the ODBC configuration files and how the DataDirect driver manager interacts with the Netezza ODBC drivers.

DataDirect Driver Manager Search Behavior

The default behavior of the DataDirect driver manager is to search the following two places, in sequence, to locate the two ODBC configuration files:

1. The driver manager looks first in the current user's home directory for the two non-hidden files **odbcinst.ini** and **odbc.ini**. If the two files exist as hidden files in the user's home directory, the driver manager ignores them.
2. If the driver manager does not find non-hidden configuration files in the user's home directory, it looks next in the **/etc** directory for the two non-hidden files **odbcinst.ini** and **odbc.ini**. If the two configuration files exist as hidden files in the **/etc** directory, the driver manager ignores them.

Incompatibilities

The following are incompatibilities between the Netezza ODBC drivers and the DataDirect driver manager:

1. The DataDirect driver manager looks first in the current user's home directory for non-hidden configuration files instead of hidden ones.
2. The DataDirect driver manager looks next in the **/etc** directory instead of in the **/usr/local/etc** directory.
3. The DataDirect driver manager also looks at the value of the environment variable **ODBCINI** to determine the location of the **odbc.ini** file to use.

To work around the DataDirect driver manager's ODBC configuration file search incompatibilities, do one of the following:

1. **Separate Configuration Files in Each Home Directory.** Use this method if you would like each user of the Linux/UNIX client system to use a separate set of ODBC configuration files. Netezza ODBC drivers look for hidden configuration files, and the DataDirect driver manager looks for non-hidden files. To overcome this incompatibility, create two soft links in each user's home directory so the DataDirect driver manager finds non-hidden files in the home directory and the Netezza ODBC driver finds hidden files:
   - A soft link so **odbcinst.ini** points to **.odbcinst.ini**
   - A soft link so **odbc.ini** points to **.odbc.ini**

2. **Shared Configuration Files.** Use this method if you would like all users of the Linux/UNIX client system to use the same set of ODBC configuration files. Configure the **NZ_ODBC_INI_PATH** environment variable so it specifies the **/etc** directory, and place the two non-hidden configuration files **odbcinst.ini** and **odbc.ini** in the **/etc** directory. The DataDirect driver manager and the Netezza ODBC driver then both find the configuration files in the **/etc** directory.
3. **Two environment variables.** Use this method if you would like both the Netezza ODBC driver and DataDirect driver manager to find the configuration files no matter where they are on the filesystem. Use both **NZ_ODBC_INI_PATH** and **ODBCINI** together to point to the **odbc.ini** and **odbcinst.ini** files anywhere on the filesystem.

Investigating Client System Configuration

Sometimes, you may need to change the configuration of a UNIX/Linux client system to access new Netezza appliance data sources after the system has previously been configured to access other types of ODBC data sources. Such a system might already have a driver manager installed and ODBC configuration files located in some directory in the file system. In such a situation, you will find it helpful to discover as much about the present configuration as you can before deciding how to configure the Netezza ODBC drivers and Netezza appliance data sources.

One technique is to use the **env** and **grep** commands to determine if there already exist one or more environment variables that point to ODBC configuration directories. You may find the following command useful:

```
env | grep -i ODBC
```

The information displayed by the above command tells you if there are any entries in the environment that contain the characters ODBC. You might see a result like the following:

```
ODBC_HOME=/opt/odbc
```

or

```
ODBCINI=/opt/odbc/odbc.ini
```

Results such as the above give you clues as to how the client system is presently configured. You can than look in the referenced directories for ODBC configuration files, and then configure the Netezza ODBC driver to look in those same directories for the ODBC configuration files.

APPENDIX B

Troubleshooting

What's in this appendix

- ODBC Debug Logging and Driver Manager Tracing
- JDBC Debug Logging
- OLE DB Debug Logging

This appendix introduces methods for tracking system activity, which can help in detecting problems and troubleshooting issues. A debug log provides you with a record of system activity, so you can take steps to correct problems.

ODBC Debug Logging and Driver Manager Tracing

This section discusses the procedures for enabling debug logging and driver manager tracing, two methods of tracking system activity. Netezza recommends you use debug logging, which provides the necessary information and is a supported driver logging function. Tracing is an external, non-Netezza driver manager option that keeps a log of function calls between the driver manager and the driver, and between the driver manager and the application. Both debugging and tracing are disabled by default and should be enabled only for troubleshooting, as they may degrade the client application performance.

ODBC Debug Logging

By default, debug logging is disabled, as enabling may affect performance.

Enabling ODBC Logging in Windows

Use the following steps to enable ODBC Logging in Windows.

1. Open the ODBC Data Source Administrator window:
   Start > Settings > Control Panel > Administrative Tools > Data Sources (ODBC)
2. In the ODBC Data Source Administrator window, select the **System DSN** tab.
3. Select the NZSQL system data source and click Configure.
4. Select the **Driver Options** tab.
5. Select Enable Debug Logging.
6. Specify the Log File Path, where the ODBC log will be saved. The log file name is nzodbc_<user><pid>.log.

7. Click OK to save your settings.

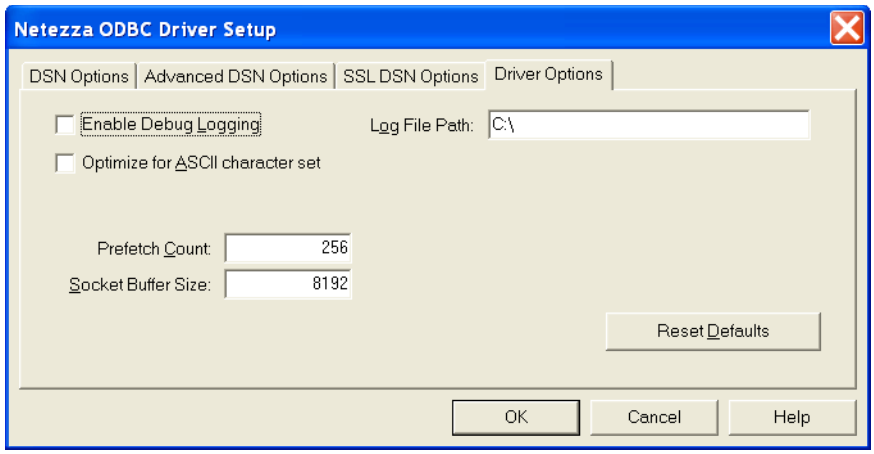

Figure B-1: Enabling Debug Logging

Enabling ODBC Logging in UNIX/Linux

Use the following steps to enable ODBC Logging in UNIX/Linux:

1. Open the .odbcinst.ini file.
2. In the NetezzaSQL section, set DebugLogging = true.
3. Save and close the file. The ODBC log will be saved in the LogPath value, and the log file name is nzodbc_<user><pid>.log.

ODBC Driver Manager Tracing

By default, tracing is disabled. Remember to disable tracing after troubleshooting, as it may degrade performance.

Enabling ODBC Driver Manager Tracing in Windows

Use the following steps to enable ODBC Driver Manager Tracing in Windows.

1. Open the ODBC Data Source Administrator window:
   Start > Settings > Control Panel > Administrative Tools > Data Sources (ODBC)
2. In the ODBC Data Source Administrator window, select the **Tracing** tab.
3. Specify the Log File Path and filename where the tracing log is saved.
4. Click Start Tracing Now and click OK.

Enabling ODBC Driver Manager Tracing for UNIX/Linux

Use the following steps to enable ODBC Driver Manager Tracing in UNIX/Linux.

1. Open the .odbcinst.ini file.
2. Make the following changes in ODBC:

   Trace = 1

   TraceAutoStop = 0

   TraceFile = /tmp
3. Save and close the file. The log file will be saved in /tmp.

JDBC Debug Logging

This section describes the debug logging mechanism for JDBC, available with the GUI, or with the following options for the connection URL:

- loglevel
- loggertype
- logdirpath

loglevel

You can set the following case-insensitive log levels:

Table B-1: Log Levels

Level	Number	Result
off	0	No messages will be logged (default).
debug	2	Logging is done at debug level. Every message will be logged.
info	3	Only information messages, entry exit statements, and fatal messages will be logged.
entry_exit	4	Only entry exit statements and fatal messages will be logged.
fatal	5	Only fatal messages will be logged.

Enter the log level either by type or by number, as in the following examples:

loglevel=entry_exit

loglevel=4

loggertype

You can set the following case-insensitive log format types:

- loggertype=text (Default)
- loggertype=html

logdirpath

You can set the directory path for the log files, as in the following:

logdirpath=<path on the local file system>

The default values for logdirpath are as follows:

- Windows: C:\
- UNIX/Linux: /tmp/

Connection String Examples

Separate connection string components with a semicolon, as in the following examples:

```
jdbc:netezza://<host>/<database>;user=<username>;password=<password>;
loglevel=<log level>;logdirpath=<log directory>
```

UNIX logdirpath:

```
jdbc:netezza://tom/will;user=admin;password=password;loglevel=2;
logdirpath=/jdbc_log_file
```

Windows logdirpath:

```
jdbc:netezza://tom/will;user=admin;password=password;loglevel=2;
logdirpath=C:\
```

OLE DB Debug Logging

OLE DB debug logging is set up via the connection URL with the following components:

- Provider — Enter NZOLEDB
- User ID — Enter your user ID
- Password — Enter your password
- Data Source — Enter the Netezza appliance hostname
- Initial Catalog — Enter the database name
- Port — Enter the port number (default is 5480)
- Logging Level — Enter 1 to enable logging or 0 for no logging
- Log Path — Enter the path for the log file (for example, c:\logs)

Separate components with a semicolon, as in the following example:

```
Provider=NZOLEDB;User ID=<UserID>;Password=<Password>;Data
Source=<npshostname>;Initial Catalog=<database name>;Port=5480;Logging
Level=1;Log Path=c:\logs
```

Index

A

architecture 1-2

C

Character Translation Option window 5-2
client tools uninstall
  Windows 3-4
configuration files. See ODBC driver or data source configuration files.
configuration. See ODBC or JDBC driver or data source configuration.

D

data source configuration. See ODBC or JDBC data source configuration.
data sources. See ODBC or JDBC data sources.
DataDirect driver manager A-3
driver configuration. See ODBC or JDBC driver configuration.
driver installation. See ODBC or JDBC driver installation.
driver manager
  absence in JDBC architecture 4-3
  DataDirect
    configuration file search behavior A-3
    Netezza ODBC driver incompatibilities A-3
  functions 4-2
  in Linux/UNIX systems 4-3
  in Windows systems 4-3
  introduction 4-2
  Netezza recommendation for Linux/UNIX systems 4-3
  unixODBC
    compatibility with Netezza ODBC drivers A-2
    configuration file search behavior A-2
    installation procedure 4-4
    installation steps 4-4

I

installation. See client tools, ODBC or JDBC installation.

J

JDBC
  configuration
    Linux 6-2
    Windows 6-3
  data source configuration
    Windows 6-3
  debug logging B-3
  driver configuration, Windows 6-3
  driver data source 2-4
  installation
    Linux 6-2
    unsupported platforms 6-2
    Windows 6-1
  online documentation Oracle Sun Developer Network URL -xv

M

MDAC 4-3
Microsoft Data Access Component (MDAC) 4-3

O

ODBC
  API 2-3
  data source configuration
    Linux 5-12
    Windows 5-3
  data source configuration file, Linux 5-12
  data source configuration procedure
    Linux 5-15
    Windows 5-4
  data source configuration steps
    Linux 5-15
  debug logging B-1
  driver configuration
    Linux 5-10
  driver configuration file, Linux 5-10
  driver configuration procedure, Linux 5-11
  driver configuration steps, Linux 5-11
  driver functions 4-1
  driver installation
    procedure for Windows 5-2
    Windows 5-1, 5-10
  driver manager tracing B-2
  driver manager. See driver manager
  driver properties 4-2
  driver version support 2-3
  installation
    Windows 5-1, 5-10
  Netezza driver version support 2-3
  Netezza-supported platforms 2-1
  online documentation Microsoft URL -xv
  standards conformance 2-3
  validating connectivity
    Linux 5-16
  validating connectivity procedure
    Linux 5-16
  validating connectivity steps
    Linux 5-16
ODBC configuration files A-1
ODBC driver
  Visual Studio 8 libraries 5-3
OLE DB
  debug logging B-4
  provider installation
    steps for Windows 7-1
  provider installation for Windows 7-2
  validating connectivity 7-3
  Visual Studio 8 libraries 7-2

Optimize for ASCII character set option 5-2

R

release compatibility 2-2

S

supported platforms 2-1

U

unixODBC
  driver manager A-2
  installation procedure 4-4
  installation steps 4-4

V

Visual Studio 8 (VC8) Libraries 5-3